Panasonic®

MICROCOMPUTER MN101E

MN101EFA8/A7/A3/A2 Series
LSI User's Manual

Pub.No. 216A8-012E

Request for your special attention and precautions in using the technical information and semiconductors described in this book

(1) If any of the products or technical information described in this book is to be exported or provided to non-residents, the laws and regulations of the exporting country, especially, those with regard to security export control, must be observed.

(2) The technical information described in this book is intended only to show the main characteristics and application circuit examples of the products. No license is granted in and to any intellectual property right or other right owned by Panasonic Corporation or any other company. Therefore, no responsibility is assumed by our company as to the infringement upon any such right owned by any other company which may arise as a result of the use of technical information described in this book.

(3) The products described in this book are intended to be used for general applications (such as office equipment, communications equipment, measuring instruments and household appliances), or for specific applications as expressly stated in this book.
Consult our sales staff in advance for information on the following applications:
・Special applications (such as for airplanes, aerospace, automotive equipment, traffic signaling equipment, combustion equipment, life support systems and safety devices) in which exceptional quality and reliability are required, or if the failure or malfunction of the products may directly jeopardize life or harm the human body.
It is to be understood that our company shall not be held responsible for any damage incurred as a result of or in connection with your using the products described in this book for any special application, unless our company agrees to your using the products in this book for any special application.

(4) The products and product specifications described in this book are subject to change without notice for modification and/or improvement. At the final stage of your design, purchasing, or use of the products, therefore, ask for the most up-to-date Product Standards in advance to make sure that the latest specifications satisfy your requirements.

(5) When designing your equipment, comply with the range of absolute maximum rating and the guaranteed operating conditions (operating power supply voltage and operating environment etc.). Especially, please be careful not to exceed the range of absolute maximum rating on the transient state, such as power-on, power-off and mode-switching. Otherwise, we will not be liable for any defect which may arise later in your equipment.
Even when the products are used within the guaranteed values, take into the consideration of incidence of break down and failure mode, possible to occur to semiconductor products. Measures on the systems such as redundant design, arresting the spread of fire or preventing glitch are recommended in order to prevent physical injury, fire, social damages, for example, by using the products.

(6) Comply with the instructions for use in order to prevent breakdown and characteristics change due to external factors (ESD, EOS, thermal stress and mechanical stress) at the time of handling, mounting or at customer's process. When using products for which damp-proof packing is required, satisfy the conditions, such as shelf life and the elapsed time since first opening the packages.

(7) This book may be not reprinted or reproduced whether wholly or partially, without the prior written permission of our company.

20100202

If you have any inquiries or questions about this book or our semiconductor products, please contact our sales division.

About This Manual

■Objective

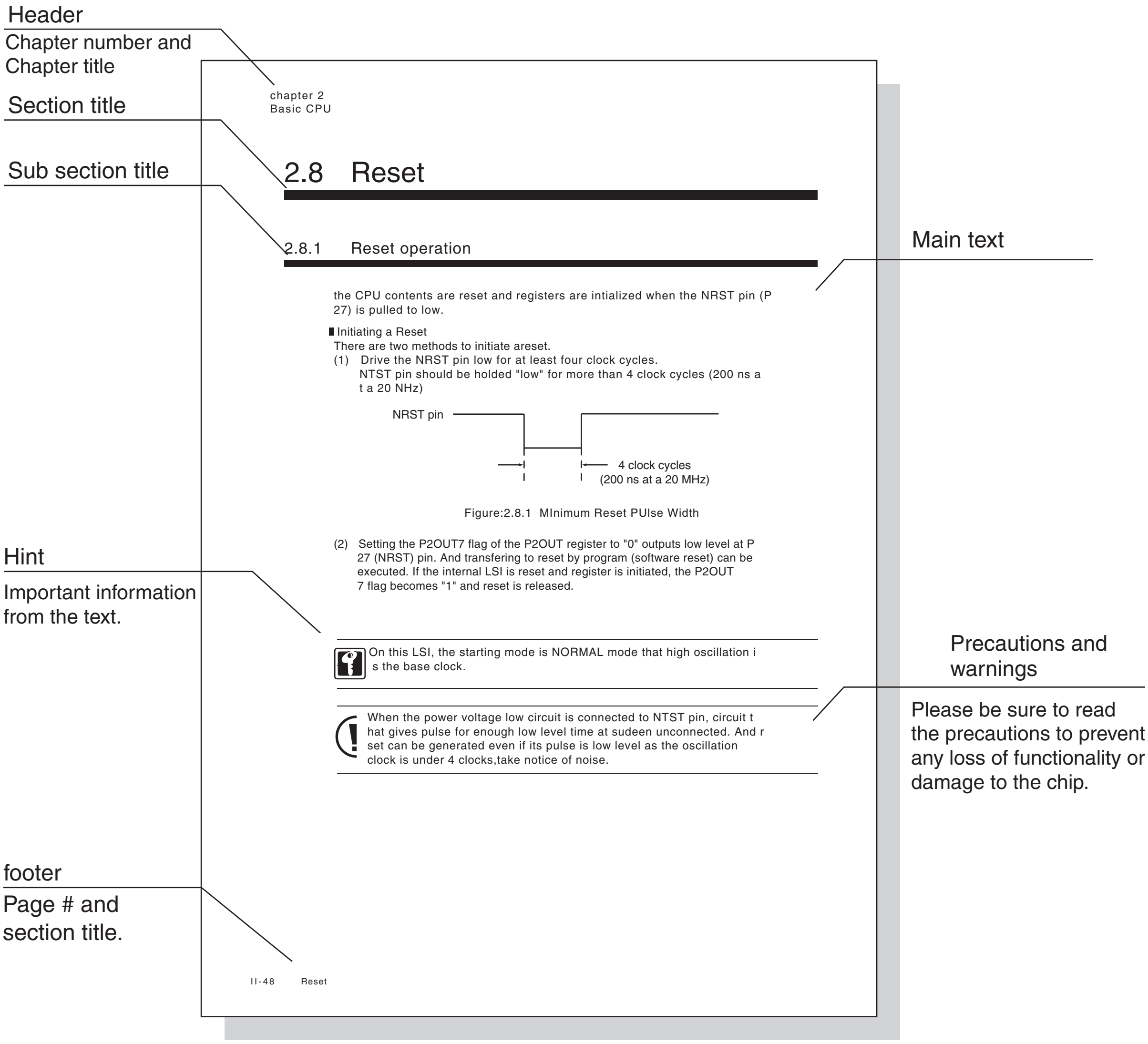
The primary objective of this LSI manual is to describe the features of this product including an overview, CPU basic functions, interrupt, port, timer, serial interface, and other peripheral hardware functions.

Each section consists of brief functional information, block diagrams and the details of control registers including operation methods and setting examples.

■Structure of This Manual

Each section of this manual consists of a title, summary, main text, hint, precautions and warnings, and references.

The layout and definition of each section are shown below.


Header
Chapter number and Chapter title
Section title
Sub section title
Main text
Hint
Important information from the text.
Precautions and warnings
Please be sure to read the precautions to prevent any loss of functionality or damage to the chip.
footer
Page # and section title.
NRST pin
4 clock cycles
(200 ns at a 20 MHz)

chapter 2
Basic CPU

2.8 Reset

2.8.1 Reset operation

the CPU contents are reset and registers are intialized when the NRST pin (P 27) is pulled to low.

■Initiating a Reset
There are two methods to initiate areset.
(1) Drive the NRST pin low for at least four clock cycles.
NTST pin should be holded "low" for more than 4 clock cycles (200 ns a t a 20 NHz)

Figure:2.8.1 MInimum Reset PUlse Width

(2) Setting the P2OUT7 flag of the P2OUT register to "0" outputs low level at P 27 (NRST) pin. And transfering to reset by program (software reset) can be executed. If the internal LSI is reset and register is initiated, the P2OUT 7 flag becomes "1" and reset is released.

On this LSI, the starting mode is NORMAL mode that high oscillation i s the base clock.

When the power voltage low circuit is connected to NTST pin, circuit t hat gives pulse for enough low level time at sudeen unconnected. And r set can be generated even if its pulse is low level as the oscillation clock is under 4 clocks,take notice of noise.

II-48 Reset

This page serves as an example to the explanations above. It may be different on an actual page.

■Finding Desired Information

This manual provides three methods for finding the desired information quickly and easily.

1.Refer to the index at the front of the manual to locate the beginning of each section.

2.Refer to the table of contents at the front of the manual to locate the desired titles.

3.The chapter number and chapter title are located at the top corner of each page, and the section titles are located at the bottom corner of each page.

■Related Manuals

Note that the following related documents are available.

- "MN101E Series Instruction Manual"
  <Describes the instruction set.>
- "MN101C/MN101E Series Cross-assembler User's Manual"
  <Describes the assembler syntax and notation.>
- "MN101C/MN101E Series C Compiler User's Manual Usage Guide"
  <Describes the installation, commands and options of the C Compiler.>
- "MN101C/MN101E Series C Compiler User's Manual Language Description"
  <Describes the syntax of the C Compiler.>
- "MN101C/MN101E Series C Compiler User's Manual Library Reference"
  <Describes the standard library of the C Compiler.>
- "MN101C/MN101E Series Installation Manual"
  <Describes the steps to install the Integrated Development Environment (DebugFactory Builder), C compiler and the real-time OS. It also describes the procedure to setup the in-circuit emulator (PX-ICE101C/E).>
- "MN101C/MN101E/MN103L Series Software Development Environment Installation Manual"
  <Describes the steps to install the Integrated Development Environment (DebugFactory Builder), C compiler and the real-time OS. It also describes the procedure to setup the in-circuit emulator (PX-ICE101C/E-Advance, PX-ICE101C/E-Lite).>

■Contact Information

Please contact our sales division.

Contents

Contents

Chapter 1　Overview

1.1 Overview

1.1.1 Overview

The MN101E series of 8-bit single-chip microcomputers (the memory expansion version of MN101C series) incorporate multiple types of peripheral functions. This chip series is well suited for automotive power window, camera, TV, CD, printer, telephone, home appliance, PPC, fax machine, music instrument and other applications.

This LSI brings to embedded microcomputer applications flexible, optimized hardware configurations and a simple efficient instruction set. MN101EFA7G/A8G/A2G/A3G have an internal 128 KB of ROM and 6 KB of RAM. MN101EFA7D/A8D/A2D/A3D have an internal 64 KB of ROM and 4 KB of RAM. Peripheral functions include 5 external interrupts, including NMI, 10 timer counters, 4 types of serial interfaces, A/D converter, watchdog timer and buzzer output. The system configuration is suitable for system control microcontroller.

With 3 oscillation systems (internal frequency: 16 MHz, high-speed crystal/ceramic frequency: max. 10 MHz, low-speed crystal/ceramic frequency: 32.768 kHz) contained on the chip, the system clock can be switched to high-speed frequency input (NORMAL mode) or PLL input (PLL mode), or low-speed frequency input (SLOW mode). The system clock is generated by dividing the oscillation clock or PLL clock. The best operation clock for the system can be selected by switching its frequency ratio by programming. High speed mode has NORMAL mode which is based on the clock dividing fpll, (fpll is generated by original oscillation and PLL), by 2 (fpll/2), and the double speed mode which is based on the clock not dividing fpll.

A machine cycle (minimum instruction execution time) in NORMAL mode is 200 ns when the original oscillation fosc is 10 MHz (PLL is not used). A machine cycle in the double speed mode, in which the CPU operates on the same clock as the external clock, is 100 ns when fosc is 10 MHz. A machine cycle in the PLL mode is 50 ns (maximum).

1.1.2 Product Summary

This manual describes the following model.

Table:1.1.1 Product Summary

Model	ROM Size	RAM Size	Classification	Capacitive Touch Detection Circuit	Package
MN101EFA8G	128 KB	6 KB	Flash EEPROM version	√	80 Pin TQFP 80 Pin LQFP
MN101EFA8D	64 KB	4 KB			
MN101EFA3G	128 KB	6 KB	Flash EEPROM version	-	
MN101EFA3D	64 KB	4 KB			
MN101EFA7G	128 KB	6 KB	Flash EEPROM version	√	64 Pin TQFP 64 Pin LQFP
MN101EFA7D	64 KB	4 KB			
MN101EFA2G	128 KB	6 KB	Flash EEPROM version	-	
MN101EFA2D	64 KB	4 KB			

1.2 Hardware Functions

■ Feature

- Memory Capacity:
  ROM 128 KB / 64 KB
  RAM 6 KB / 4 KB

- Package:
  MN101EFA8/A3 Series
  80-Pin TQFP (12 mm × 12 mm / 0.50 mm pitch)
  80-Pin LQFP (14 mm × 14 mm / 0.65 mm pitch)

  MN101EFA7/A2 Series
  64-Pin TQFP (10 mm × 10 mm / 0.50 mm pitch)
  64-Pin LQFP (14 mm × 14 mm / 0.80 mm pitch)

- Machine Cycle:
  High-speed mode 0.05 µs / 20 MHz (4.0 V to 5.5 V)
  Low-speed mode 62.5 µs / 32 kHz (4.0 V to 5.5 V)

- Oscillation circuit: 3 channel oscillation circuit
  Internal oscillation (frc): 16 MHz
  Crystal/ceramic (fosc): Maximum 10 MHz
  Crystal/ceramic (fx): Maximum 32.768 kHz

-Clock Multiplication circuit (PLL Circuit)
PLL circuit output clock (fpll): fosc multiplied by 2, 3, 4, 5, 6, 8, 10,
1/2 × frc multiplication by 4, 5 enable

-Clock Gear for System Clock
System Clock (fs): fpll divided by 1, 2, 4, 16, 32, 64, 128

-Clock Gear for control clock of peripheral function
Control clock of peripheral function (fpll-div): stop or fpll divided by 1, 2, 4, 8, 16

- Memory Bank:
  Expands data memory space by the bank system (by 64 KB, 16 banks)
  Source address bank / Destination address bank

- Operation Mode:
  NORMAL mode (High-speed mode)
  SLOW mode (Low-speed mode)
  HALT mode
  STOP mode
  (The operation clock can be switched in each mode.)

- Operating Voltage: 4.0 V to 5.5 V

- Operation ambient temperature: -40 °C to +85 °C

- Interrupt:
  MN101EFA8 Series: 36 interrupts
  MN101EFA3 Series: 28 interrupts
  MN101EFA7 Series: 32 interrupts
  MN101EFA2 Series: 28 interrupts

  <Non-maskable interrupt>
    - Non-maskable interrupt and Watchdog timer overflow interrupt

  <Timer interrupts>
    - Timer 0 interrupt
    - Timer 1 interrupt
    - Timer 2 interrupt
    - Timer 3 interrupt
    - Timer 6 interrupt
    - Time base timer interrupt
    - Timer 7 interrupt
    - Timer 7 compare register 2 match interrupt
    - Timer 8 interrupt
    - Timer 8 compare register 2 match interrupt
    - Timer 9 overflow interrupt
    - Timer 9 underflow interrupt
    - Timer 9 compare register 2 match interrupt

  <Serial Interface interrupts>
    - Serial interface 0 interrupt
    - Serial interface 0 UART reception interrupt
    - Serial interface 1 interrupt
    - Serial interface 1 UART reception interrupt
    - Serial interface 2 interrupt
    - Serial interface 2 UART reception interrupt
    - Serial interface 4 interrupt
    - Serial interface 4 stop condition interrupt

  <A/D interrupt>
    - A/D conversion interrupt

  <External interrupts>
    - IRQ0: Edge selectable, noise filter connection available
    - IRQ1: Edge selectable, noise filter connection available
    - IRQ2: Edge selectable, noise filter connection available, both edges interrupt
    - IRQ3: Edge selectable, noise filter connection available, both edges interrupt
    - IRQ4: Edge selectable, noise filter connection available, both edges interrupt, Key scan interrupt

<Touch Detect interrupts>

- Touch 0 detect interrupt
- Touch 0 detect error interrupt
- Touch 0 round interrupt
- Touch 0 data transmission interrupt
  (MN101EFA3/A2 Series don't have this function)

- Touch 1 detect interrupt
- Touch 1 detect error interrupt
- Touch 1 round interrupt
- Touch 1 data transmission interrupt
  (MN101EFA7/A3/A2 Series don't have this function)

- Timer Counter: 10 timers
  - 8-bit timer for general use × 4 sets
  - 16-bit timer for general use × 2 sets
  - Motor control 16-bit timer × 1 set
  - 8-bit free-run timer × 1 set
  - Time base timer × 1 set
  - Baud rate timer × 1 set

Timer 0 (8-bit timer for general use)

- Square wave output (Timer pulse output)
- Added pulse (2-bit) type PWM output can be output to large current pin TM0IOA
- Event count
- Simple pulse measurement
- Clock source
  fpll-div, fpll-div/4, fpll-div/16, fpll-div/32, fpll-div/64, fpll-div/128, fs/2, fs/4, fs/8, fx, External clock, Timer A output

Timer 1 (8-bit timer for general use)

- Square wave output (Timer pulse output) can be output to large current pin TM1IOA
- Event count
- 16-bit cascade connected (with Timer 0)
- Clock source
  fpll-div, fpll-div/4, fpll-div/16, fpll-div/32, fpll-div/64, fpll-div/128, fs/2, fs/4, fs/8, fx, External clock, Timer A output

Timer 2 (8-bit timer for general use)

- Square wave output (Timer pulse output)
- Added pulse (2-bit) type PWM output can be output to large current pin TM2IOA
- Event count
- Simple pulse measurement
- 24-bit cascade connected (with Timer 0 and Timer 1)
- Clock source
  fpll-div, fpll-div/4, fpll-div/16, fpll-div/32, fpll-div/64, fpll-div/128, fs/2, fs/4, fs/8, fx, External clock, Timer A output

Timer 3 (8-bit timer for general use)

- Square wave output (Timer pulse output) can be output to large current pin TM3IOA
- Event count
- 16-bit cascade connected (with Timer 2)
- 32-bit cascade connected (with Timer 0 and Timer 1 and Timer 2)
- Clock source

  fpll-div, fpll-div/4, fpll-div/16, fpll-div/32, fpll-div/128, fs/2, fs/4, fs/8, fx,
  External clock, Timer A output

Timer 6 (8-bit free-run timer, Time base timer)

8-bit free-run timer

- Clock source

  fpll-div, fpll-div/2^{12}, fpll-div/2^{13}, fs, fx, fx/2^2, fx/2^3, fx/2^{12}, fx/2^{13}

Time base timer

- Interrupt generation cycle

  fpll-div/2^7, fpll-div/2^8, fpll-div/2^9, fpll-div/2^{10}, fpll-div/2^{13}, fpll-div/2^{15}, fx/2^7, fx/2^8, fx/2^9, fx/2^{10}, fx/2^{13}, fx/2^{15}

Timer 7 (16-bit timer for general use)

- Square wave output (Timer pulse output)
- High precision PWM output (Cycle/Duty continuous changeable) can be output to large current pin TM7IOA
- Event count
- Input capture function (Both edges can be operated)
- Clock source

  fpll-div, fpll-div/2, fpll-div/4, fpll-div/16, fs, fs/2, fs/4, fs/16,
  Timer A divided by 1, 2, 4, 16, External clock divided by 1, 2, 4, 16

Timer 8 (16-bit timer for general use)

- Square wave output (Timer pulse output)
- High precision PWM output (Cycle/Duty continuous changeable) can be output to large current pin TM8IOA
- Event count
- Input capture function (Both edges can be operated)
- Clock source

  fpll-div, fpll-div/2, fpll-div/4, fpll-div/16, fs, fs/2, fs/4, fs/16,
  Timer A divided by 1, 2, 4, 16, External clock divided by 1, 2, 4, 16

Timer 9 (Motor control 16-bit timer)

- Square wave output (Timer pulse output) can be output to large current pin TM9IOA
- Event count
- Complementary 3-phase PWM output can be output to large current pin TM9OD0 to TM9OD5

  (Triangle wave and saw tooth wave are supported, dead time insertion available)
- Clock source

  fpll-div, fpll-div/2, fpll-div/4, fpll-div/16, fs, fs/2, fs/4, fs/16,
  Timer A divided by 1, 2, 4, 16, External clock divided by 1, 2, 4, 16

Timer A (Baud rate timer)

- Clock output for peripheral functions
- Clock source

  fpll-div, fpll-div/2, fpll-div/4, fpll-div/8, fpll-div/16, fpll-div/32, fs/2, fs/4

- Watchdog timer

  Time-out cycle can be selected from fs/2^{16}, fs/2^{18}, fs/2^{20}
  On detection of 2 errors, forcibly hard reset inside LSI.
  Operation start timing is selectable. (At reset release or write to register)

- Buzzer Output/ Reverse Buzzer Output

  Output frequency can be selected from fpll-div/2^{9}, fpll-div/2^{10}, fpll-div/2^{11}, fpll-div/2^{12}, fpll-div/2^{13}, fpll-div/2^{14}, fx/2^{3}, fx/2^{4}

- A/D Converter: 10-bit × 16 channels (MN101EFA8/A3 Series)
  10-bit × 12 channels (MN101EFA7/A2 Series)

- Serial Interface: 4 channels

  Serial 0: UART (full duplex)/ Clock synchronous
  - Clock synchronous serial interface
    - Transfer clock source fpll-div/2, fpll-div/4, fpll-div/16, fpll-div/64, fs/2, fs/4, Timer 0 to 3 or Timer A divided by 1, 2, 4, 8, 16, External clock
    - MSB/LSB can be selected as the first bit to be transferred, arbitrary sizes of 2 to 8 bits are selectable.
    - Sequence transmission, reception or both are available
  - Full duplex UART
    - Baud rate timer, selected from Timer 0 to 3 or Timer A
    - Parity check, overrun error/ framing error detection
    - Transfer size 7 to 8 bits can be selected

  Serial 1: UART (full duplex)/ Clock synchronous
  - Clock synchronous serial interface
    - Transfer clock source fpll-div/2, fpll-div/4, fpll-div/16, fpll-div/64, fs/2, fs/4, Timer 0 to 3 or Timer A divided by 1, 2, 4, 8, 16, External clock
    - MSB/LSB can be selected as the first bit to be transferred, arbitrary sizes of 2 to 8 bits are selectable.
    - Sequence transmission, reception or both are available.
  - Full duplex UART
    - Baud rate timer, selected from Timer 0 to 3 or Timer A
    - Parity check, overrun error/ framing error detection
    - Transfer size 7 to 8 bits can be selected

  Serial 2: UART (full duplex)/ Clock synchronous
  - Clock synchronous serial interface
    - Transfer clock source fpll-div/2, fpll-div/4, fpll-div/16, fpll-div/64, fs/2, fs/4, Timer 0 to 3 or Timer A divided by 1, 2, 4, 8, 16, External clock
    - MSB/LSB can be selected as the first bit to be transferred, arbitrary sizes of 2 to 8 bits are selectable.
    - Sequence transmission, reception or both are available.
  - Full duplex UART
    - Baud rate timer, selected from Timer 0 to 3 or Timer A
    - Parity check, overrun error/ framing error detection
    - Transfer size 7 to 8 bits can be selected

Serial 4: Multi master IIC/ Clock synchronous

Clock synchronous serial interface

- Transfer clock source fpll-div/2, fpll-div/4, fpll-div/16, fpll-div/32, fs/2, fs/4, Timer 0 to 3 or Timer A divided by 1, 2, 4, 8, 16, External clock
- MSB/LSB can be selected as the first bit to be transferred, arbitrary sizes of 2 to 8 bits are selectable.
- Sequence transmission, reception or both are available.

Multi master IIC

- 7-bit slave address is settable.
- General call communication mode is supported.

- Automatic Reset:

  Power detection level: 4.3 V (at rising), 4.2 V (at falling)

- LED Driver: 8 pins (Port A)

- Touch Sensor Timer: 2 unit/ 12 channels (MN101EFA8 Series only)
  1 unit/ 8 channels (MN101EFA7 Series only)

- Ports (MN101EFA8/A3 Series)

I/O ports	70 pins
Serial Interface pins	21 pins
Timer I/O	19 pins
Buzzer output pins	4 pins
A/D input pins	16 pins
External Interrupt pins	5 pins
LED (large current) driver	8 pins
Touch sensor input pins	12 pins (MN101EFA3 Series does not have this function)
Touch sensor resistor connect pins	4 pins (MN101EFA3 Series does not have this function)
High-speed oscillation	2 pins
Low-speed oscillation	2 pins
Special pins	9 pins
Operation mode input pins	3 pins
Reset input pin	1 pin
Analog reference voltage input pin	1 pin
Power pins	4 pins

- Ports (MN101EFA7/A2 Series)

I/O ports	55 pins
Serial Interface pins	15 pins
Timer I/O	9 pins
Buzzer output	4 pins
A/D input pins	12 pins
External Interrupt pins	5 pins
LED (large current) driver	8 pins
Touch sensor input pins	8 pins (MN101EFA2 Series does not have this function)
Touch sensor resistor connect pins	2 pins (MN101EFA2 Series does not have this function)
High-speed oscillation	2 pins
Low-speed oscillation	2 pins

Special pins	8 pins
Operation mode input pins	3 pins
Reset input pin	1 pin
Analog reference voltage input pin	1 pin
Power pins	3 pins

1.3 Pin Description

1.3.1 Pin configuration

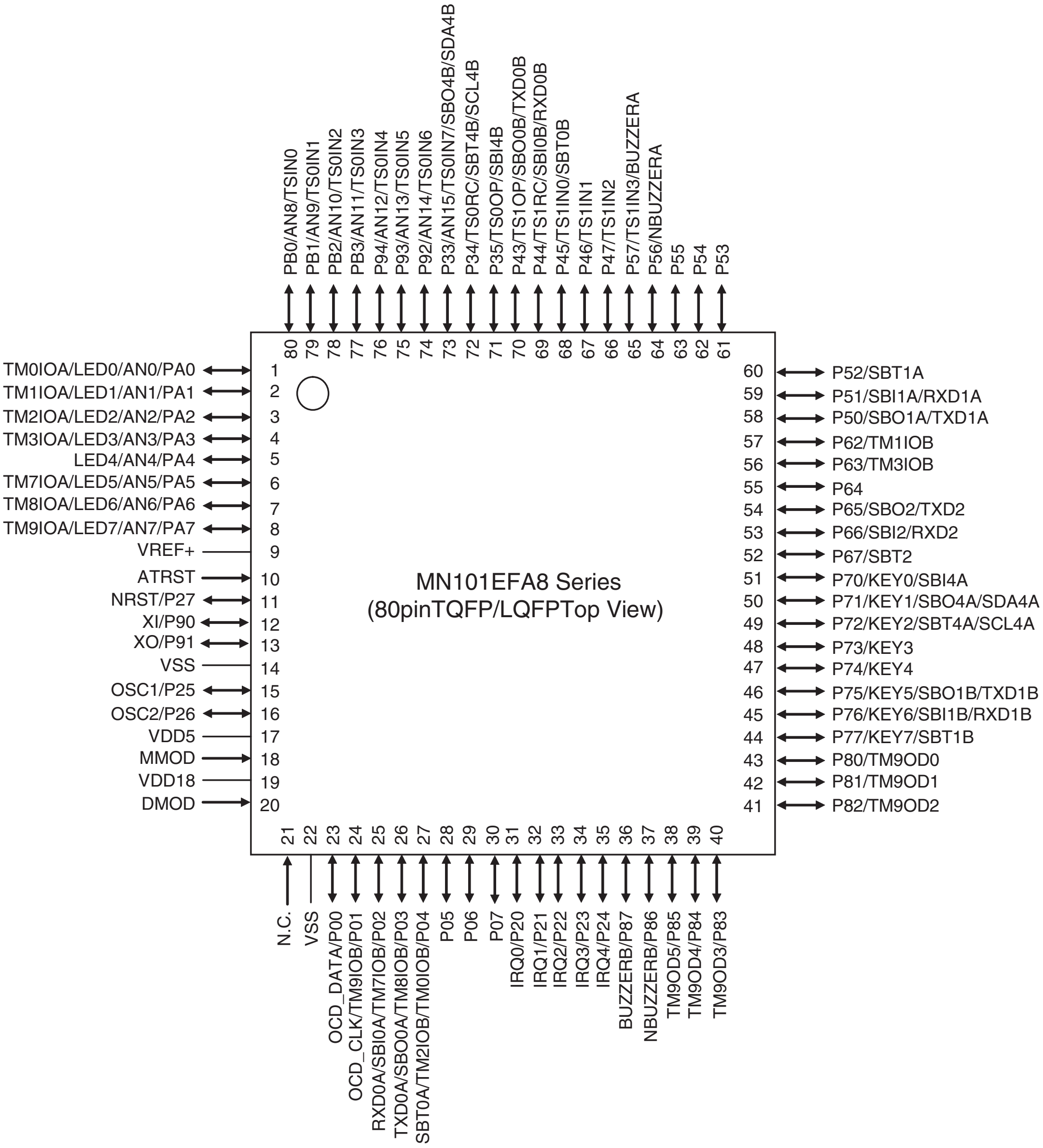


Figure:1.3.1 Pin Configuration (MN101EFA8 Series 80-pin TQFP/LQFP)

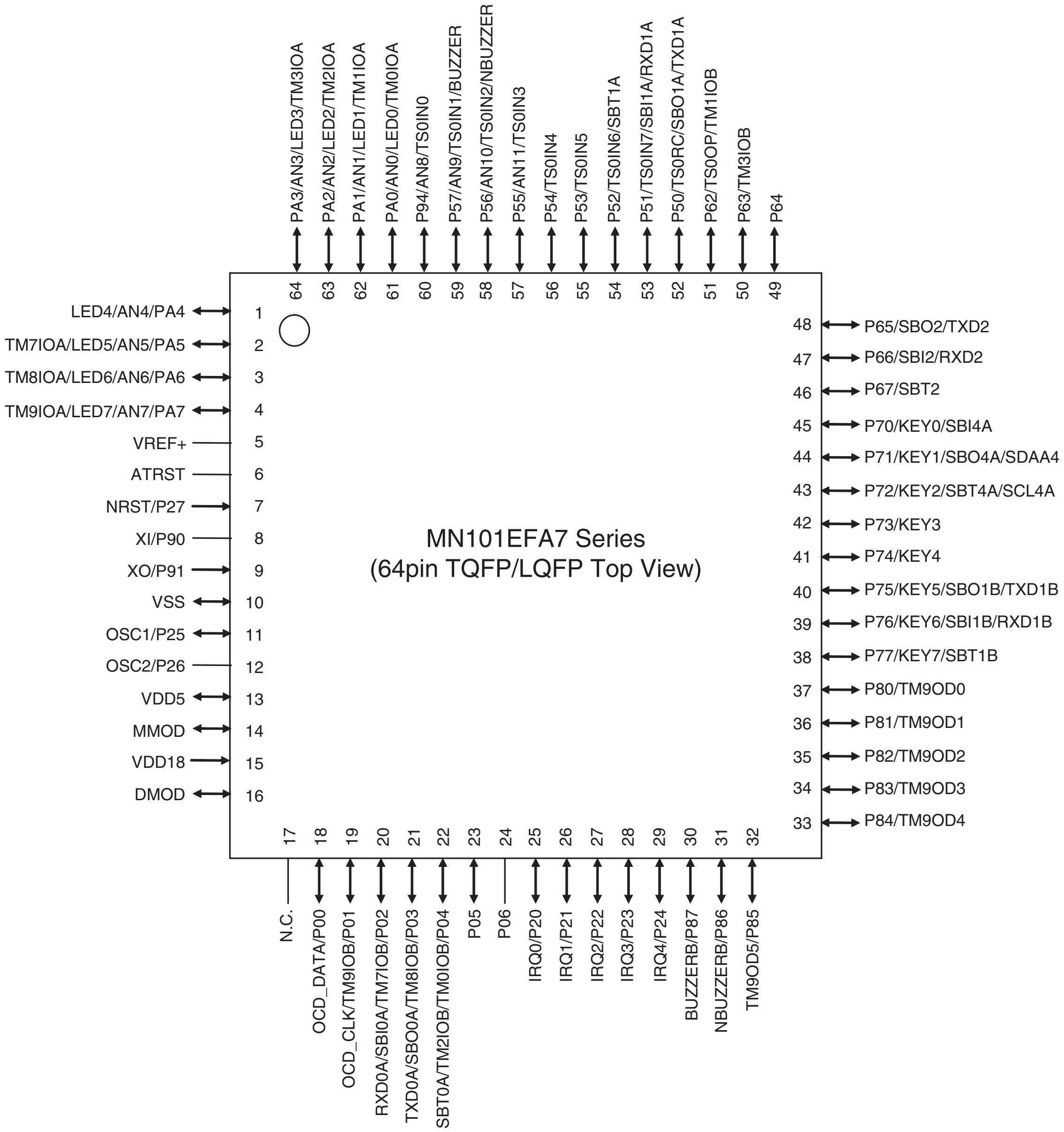


Figure:1.3.2 Pin Configuration (MN101EFA7 Series 64-pin TQFP/LQFP)

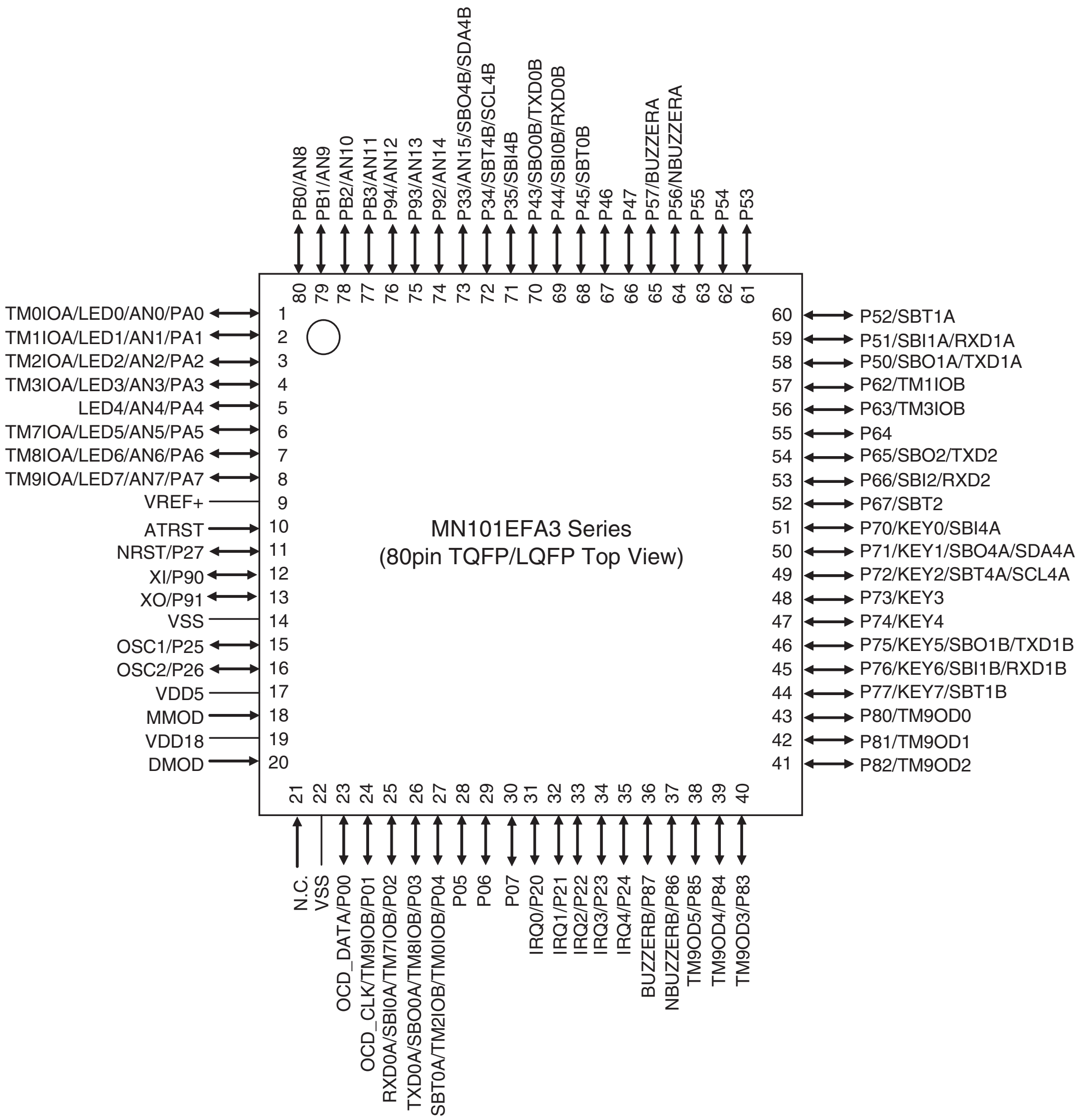


Figure:1.3.3 Pin Configuration (MN101EFA3 Series 80-pin TQFP/LQFP)

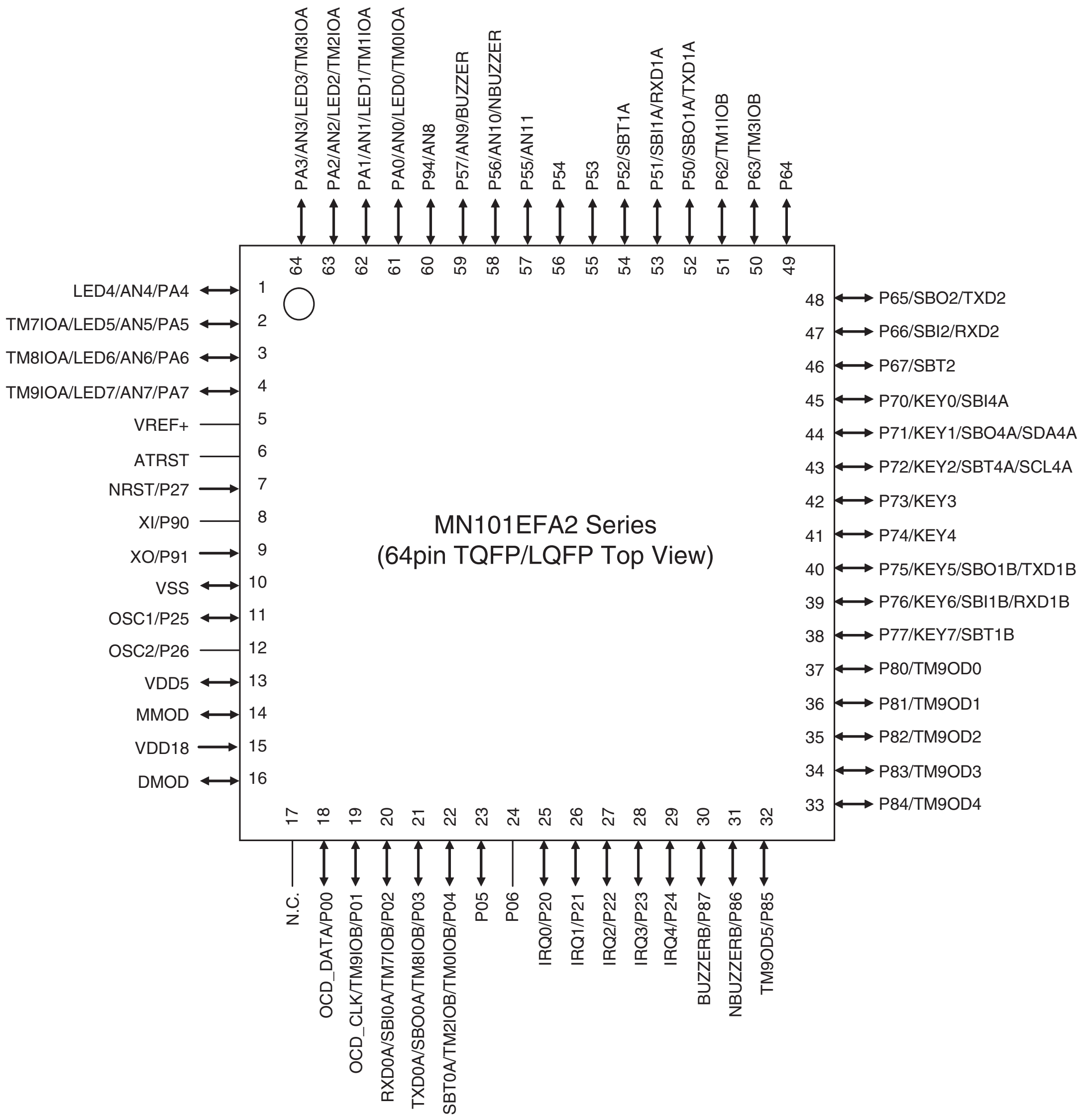


Figure:1.3.4 Pin Configuration (MN101EFA2 Series 64-pin TQFP/LQFP)

1.3.2 Pin Specification

Table remarks √: With function -: Without function

Pins	I/O	Direction Control	Pin Control	Special Functions	Functions Description	MN101EFA8 Series	MN101EFA3 Series	MN101EFA7 Series	MN101EFA2 Series
P00	in/out	P0DIR0	P0PLU0	OCD_DATA	On-boad programmer data pin	√	√	√	√
P01	in/out	P0DIR1	P0PLU1	TM9IOB	Timer 9 input/output	√	√	√	√
				OCD_CLK	On-boad programmer clock supply pin				
P02	in/out	P0DIR2	P0PLU2	TM7IOB	Timer 7 input/output	√	√	√	√
				SBI0A	Serial 0 data input				
				RXD0A	UART 0 data input				
P03	in/out	P0DIR3	P0PLU3	TM8IOB	Timer 8 input/output	√	√	√	√
				SBO0A	Serial 0 data input/output				
				TXD0A	UART 0 data input/output				
P04	in/out	P0DIR4	P0PLU4	TM0IOB	Timer 0 input/output	√	√	√	√
				TM2IOB	Timer 2 input/output				
				SBT0A	Serial 0 clock input/output				
P05	in/out	P0DIR5	P0PLU5	-	-	√	√	√	√
P06	in/out	P0DIR6	P0PLU6	-	-	√	√	√	√
P07	in/out	P0DIR7	P0PLU7	-	-	√	√	-	-
P20	in/out	P2DIR0	P2PLU0	IRQ0	External Interrupt 0	√	√	√	√
P21	in/out	P2DIR1	P2PLU1	IRQ1	External Interrupt 1	√	√	√	√
P22	in/out	P2DIR2	P2PLU2	IRQ2	External Interrupt 2	√	√	√	√
P23	in/out	P2DIR3	P2PLU3	IRQ3	External Interrupt3	√	√	√	√
P24	in/out	P2DIR4	P2PLU4	IRQ4	External Interrupt4	√	√	√	√
P25	in/out	P2DIR5	P2PLU5	OSC1	Seramic/crystal high-speed clock input	√	√	√	√
P26	in/out	P2DIR6	P2PLU6	OSC2	Seramic/crystal high-speed clock output	√	√	√	√
P27	in/out	-	-	NRST	Reset	√	√	√	√
P33	in/out	P3DIR3	P3PLUD3	SB04B	Serial 4 data input/output	√	√	-	-
				SDA4B	Multi-master IIC 4 data input/output				
				AN15	Analog 15 input				
				TS0IN7	Touch sensor 0 input 7		-		
P34	in/out	P3DIR4	P3PLUD4	SBT4B	Serial 4 clock input/output	√	√	-	-
				SCL4B	Multi-master IIC 4 clock input/output				
				TS0RC	Touch sensor 0 RC connect		-		
P35	in/out	P3DIR5	P3PLUD5	SBI4B	Serial 4 data input	√	√	-	-
				TS0OP	Touch sensor 0 output		-		
P43	in/out	P4DIR3	P4PLU3	SBO0B	Serial 0 data input/output	√	√	-	-
				TXD0B	UART 0 data input/output				
				TS1OP	Touch sensor 1 output		-		
P44	in/out	P4DIR4	P4PLU4	SBI0B	Serial 0 data input	√	√	-	-
				RXD0B	UART 0 data input				
				TS1RC	Touch sensor 1 RC connect		-		

Table remarks √: With function -: Without function

Pins	I/O	Direction Control	Pin Control	Special Functions	Functions Description	MN101EFA8 Series	MN101EFA3 Series	MN101EFA7 Series	MN101EFA2 Series
P45	in/out	P4DIR5	P4PLU5	SBT0B	Serial 0 clock input/output	√	√	-	-
				TS1IN0	Touch sensor 1 input 0		-		
P46	in/out	P4DIR6	P4PLU6	-	-	√	√	-	-
				TS1IN1	Touch sensor 1 input 1		-		
P47	in/out	P4DIR7	P4PLU7	-	-	√	√	-	-
				TS1IN2	Touch sensor 1 input 2		-		
P50	in/out	P5DIR0	P5PLU0 *1 P5PLUD0 *2	SBO1A	Serial 1 data input/output	√	√	√	√
				TXD1A	UART 1 data input/output				
				TS0RC	Touch sensor 0 RC connect	-	-	√	-
P51	in/out	P5DIR1	P5PLU1 *1 P5PLUD1 *2	SBI1A	Serial 1 data input	√	√	√	√
				RXD1A	UART 1 data input				
				TS0IN7	Touch sensor 0 input 7	-	-		-
P52	in/out	P5DIR2	P5PLU2 *1 P5PLUD2 *2	SBT1A	Serial 1 clock input/output	√	√	√	√
				TS0IN6	Touch sensor 0 input 6	-	-		-
P53	in/out	P5DIR3	P5PLU3 *1 P5PLUD3 *2	-	-	√	√	√	√
				TS0IN5	Touch sensor 0 input 5	-	-		-
P54	in/out	P5DIR4	P5PLU4 *1 P5PLUD4 *2	-	-	√	√	√	√
				AN11	Analog 11 input	-	-		
				TS0IN4	Touch sensor 0 input 4				-
P55	in/out	P5DIR5	P5PLU5 *1 P5PLUD5 *2	-	-	√	√	√	√
				AN10	Analog 10 input	-	-		
				TS0IN3	Touch sensor 0 input 3				-
P56	in/out	P5DIR6	P5PLU6 *1 P5PLUD6 *2	NBUZZERA	Buzzer reverse output	√	√	√	√
				AN9	Analog 9 input	-	-		
				TS0IN2	Touch sensor 0 input 2				-
P57	in/out	P5DIR7	P5PLU7 *1 P5PLUD7 *2	BUZZERA	Buzzer output	√	√	√	√
				TS1IN3	Touch sensor 1 input 3		-	-	-
				TS0IN1	Touch sensor 0 input 1	-		√	
P62	in/out	P6DIR2	P6PLU2	TM1IOB	Timer 1 input/output	√	√	√	√
				TS0OP	Touch sensor 0 output	-	-		-
P63	in/out	P6DIR3	P6PLU3	TM3IOB	Timer 3 input/output	√	√	√	√
P64	in/out	P6DIR4	P6PLU4	-	-	√	√	√	√
P65	in/out	P6DIR5	P6PLU5	SBO2	Serial 2 data input/output	√	√	√	√
				TXD2	UART 2 data input/output				
P66	in/out	P6DIR6	P6PLU6	SBI2	Serial 2 data input	√	√	√	√
				RXD2	UART 2 data input				
P67	in/out	P6DIR7	P6PLU7	SBT2	Serial 2 clock input/output	√	√	√	√
P70	in/out	P7DIR0	P7PLU0	KEY0	Key interrupt 0	√	√	√	√
				SBI4A	Serial 4 data input				

*1 MN101EFA8/A3 Series

*2 MN101EFA7/A2 Series

Table remarks √: With function -: Without function

Pins	I/O	Direction Control	Pin Control	Special Functions	Functions Description	MN101EFA8 Series	MN101EFA3 Series	MN101EFA7 Series	MN101EFA2 Series
P71	in/out	P7DIR1	P7PLU1	KEY1	Key interrupt 1	√	√	√	√
				SBO4A	Serial 4 data input/output				
				SDA4A	Multi-master IIC 4 data input/output				
P72	in/out	P7DIR2	P7PLU2	KEY2	Key interrupt 2	√	√	√	√
				SBT4A	Serial 4 clock input/output				
				SCL4A	Multi-master IIC 4 clock input/output				
P73	in/out	P7DIR3	P7PLU3	KEY3	Key interrupt 3	√	√	√	√
P74	in/out	P7DIR4	P7PLU4	KEY4	Key interrupt 4	√	√	√	√
P75	in/out	P7DIR5	P7PLU5	KEY5	Key interrupt 5	√	√	√	√
				SBO1B	Serial 1 data input/output				
				TXD1B	UART 1 data input/output				
P76	in/out	P7DIR6	P7PLU6	KEY6	Key interrupt 6	√	√	√	√
				SBI1B	Serial 1 data input				
				RXD1B	UART 1 data input				
P77	in/out	P7DIR7	P7PLU7	KEY7	Key interrupt 7	√	√	√	√
				SBT1B	Serial 1 clock input/output				
P80	in/out	P8DIR0	P8PLU0	TM9OD0	Timer 9 output 0	√	√	√	√
P81	in/out	P8DIR1	P8PLU1	TM9OD1	Timer 9 output 1	√	√	√	√
P82	in/out	P8DIR2	P8PLU2	TM9OD2	Timer 9 output 2	√	√	√	√
P83	in/out	P8DIR3	P8PLU3	TM9OD3	Timer 9 output 3	√	√	√	√
P84	in/out	P8DIR4	P8PLU4	TM9OD4	Timer 9 output 4	√	√	√	√
P85	in/out	P8DIR5	P8PLU5	TM9OD5	Timer 9 output 5	√	√	√	√
P86	in/out	P8DIR6	P8PLU6	NBUZZERB	Buzzer reverse output	√	√	√	√
P87	in/out	P8DIR7	P8PLU7	BUZZERB	Buzzer output	√	√	√	√
P90	in/out	P9DIR0	P9PLUD0	XI	Seramic/crystal low-speed clock input	√	√	√	√
P91	in/out	P9DIR1	P9PLUD1	XO	Seramic/crystal low-speed clock output	√	√	√	√
P92	in/out	P9DIR2	P9PLUD2	AN14	Analog 14 input	√	√	-	-
				TS0IN6	Touch sensor 0 input 6		-		
P93	in/out	P9DIR3	P9PLUD3	AN13	Analog 13 input	√	√	-	-
				TS0IN5	Touch sensor 0 input 5		-		
P94	in/out	P9DIR4	P9PLUD4	AN12	Analog 12 input	√	√	-	-
				TS0IN4	Touch sensor 0 input 4		-		
				AN8	Analog 8 input	-		√	√
				TS0IN0	Touch sensor 0 input 0				-
PA0	in/out	PADIR0	PAPLU0	AN0	Analog 0 input	√	√	√	√
				LED0	LED driving pin 0				
				TM0IOA	Timer 0 input/output				

Table remarks √: With function -: Without function

Pins	I/O	Direction Control	Pin Control	Special Functions	Functions Description	MN101EFA8 Series	MN101EFA3 Series	MN101EFA7 Series	MN101EFA2 Series
PA1	in/out	PADIR1	PAPLU1	AN1	Analog 1 input	√	√	√	√
				LED1	LED driving pin 1				
				TM1IOA	Timer 1 input/output				
PA2	in/out	PADIR2	PAPLU2	AN2	Analog 2 input	√	√	√	√
				LED2	LED driving pin 2				
				TM2IOA	Timer 2 input/output				
PA3	in/out	PADIR3	PAPLU3	AN3	Analog 3 input	√	√	√	√
				LED3	LED driving pin 3				
				TM3IOA	Timer 3 input/output				
PA4	in/out	PADIR4	PAPLU4	AN4	Analog 4 input	√	√	√	√
				LED4	LED driving pin 4				
PA5	in/out	PADIR5	PAPLU5	AN5	Analog 5 input	√	√	√	√
				LED5	LED driving pin 5				
				TM7IOA	Timer 7 input/output				
PA6	in/out	PADIR6	PAPLU6	AN6	Analog 6 input	√	√	√	√
				LED6	LED driving pin 6				
				TM8IOA	Timer 8 input/output				
PA7	in/out	PADIR7	PAPLU7	AN7	Analog 7 input	√	√	√	√
				LED7	LED driving pin 7				
				TM9IOA	Timer 9 input/output				
PB0	in/out	PBDIR0	PBPLUD0	AN8	Analog 8 input	√	√	-	-
				TS0IN0	Touch sensor 0 input 0		-		
PB1	in/out	PBDIR1	PBPLUD1	AN9	Analog 9 input	√	√	-	-
				TS0IN1	Touch sensor 0 input 1		-		
PB2	in/out	PBDIR2	PBPLUD2	AN10	Analog 10 input	√	√	-	-
				TS0IN2	Touch sensor 0 input 2		-		
PB3	in/out	PBDIR3	PBPLUD3	AN11	Analog 11 input	√	√	-	-
				TS0IN3	Touch sensor 0 input 3		-		

1.3.3 Pin Functions

Table remarks -: Without function

Pins	MN101EF A8/A3 Series	MN101EF A7/A2 Series	I/O	Function	Description
VDD5	17	13	-	Power connect pins	Apply 4.0 V to 5.5 V to VDD5 and 0 V connect 0.1 μF + 1 μF or larger bypass capacitor for internal power stabilization.
VSS	14, 22	10	-		
VDD18	19	15	-	Internal power output pin	This pin is output 1.8 V from internal power circuit. Don’t use the power supply to external device. For internal power circuit output stability, connect at least 0.1 μF + 1 μF one bypass capacitor between VDD18 and VSS.
OSC1	15	11	Input	High speed operation clock input pin	Connect these oscillation pins to ceramic or crystal ocsillators for high-frequency clock operation. If the clock is an external input, connect it to OSC1 and leave OSC2 open. The chip will not operate with an external clock when using STOP mode.
OSC2	16	12	Output	High speed operation clock output pin	
NRST	11	7	I/O	Reset pin [Active low]	This pin resets the chip when power is turned on, is allocated as P27 and contains an internal pull-up resistor (Typ. 50 kΩ). Setting this pin low initialize the internal state of the device. Thereafter, setting the input to high releases the reset. The hardware waits for the system clock to stabilize, then processes the reset interrupt. If a capacitor is to be inserted between NRST and VSS, it is recommended that a discharge diode be placed between NRST and VDD5.
ATRST	10	6	input	Auto reset setting pin	Input "High" to enable auto reset function and "Low” to disable this function
P00	23	18	I/O	I/O port 0	8-bit CMOS tri-state I/O port. Each bit can be set individually as either an input or output by P0DIR register. A pull-up resistor for each bit can be selected individually by P0PLU register. At reset, the input mode is selected and pull-up resistor is disabled (high impedance).
P01	24	19			
P02	25	20			
P03	26	21			
P04	27	22			
P05	28	23			
P06	29	24			
P07	30	-			
P20	31	25	I/O	I/O port 2	7-bit CMOS tri-state I/O port. Each bit can be set individually as either an input or output by P2DIR register. A pull-up resistor for each bit can be selected individually by P2PLU register. At reset, the input mode is selected and pull-up resistor is disabled (high impedance)
P21	32	26			
P22	33	27			
P23	34	28			
P24	35	29			
P25	15	11			
P26	16	12			
P27	11	7	input	input port 2	P27 has an N-channel open-drain configuration.
P33	73	-	I/O	I/O port 3	3-bit CMOS tri-state I/O port. Each bit can be set individually as either an input or output by P3DIR register. A pull-up /pull-down resistor for each bit can be selected individually by P3PLUD register. A pull-up/down resistor connection for each port can be selected individually in SELUD register. A pull-up/pull down can not be mixed. At reset, the input mode is selected and pull-up resistor is disabled (high impedance).
P34	72	-			
P35	71	-			
P43	70	-	I/O	I/O port 4	5-bit CMOS tri-state I/O port. Each bit can be set individually as either an input or output by P4DIR register. A pull-up resistor for each bit can be selected individually by P4PLU register. At reset, the input mode is selected and pull-up resistor is disabled (high impedance).
P44	69	-			
P45	68	-			
P46	67	-			
P47	66	-			

Table remarks -: Without function

Pins	MN101EF A8/A3 Series	MN101EF A7/A2 Series	I/O	Function	Description
P50	58	52	I/O	I/O port 5	8-bit CMOS tri-state I/O port. Each bit can be set individually as either an input or output by P5DIR register. A pull-up /pull-down resistor for each bit can be selected individually by P5PLUD register. A pull-up/down resistor connection for each port can be selected individually in SELUD register. A pull-up/pull down can not be mixed. At reset, the input mode is selected and pull-up resistor is disabled (high impedance). Pull-down function is not equipped in MN101EFA8/A3 Series.
P51	59	53			
P52	60	54			
P53	61	55			
P54	62	56			
P55	63	57			
P56	64	58			
P57	65	59			
P62	57	51	I/O	I/O port 6	6-bit CMOS tri-state I/O port. Each bit can be set individually as either an input or output by P6DIR register. A pull-up resistor for each bit can be selected individually by P6PLU register. At reset, the input mode is selected and pull-up resistor is disabled (high impedance).
P63	56	50			
P64	55	49			
P65	54	48			
P66	53	47			
P67	52	46			
P70	51	45	I/O	I/O port 7	8-bit CMOS tri-state I/O port. Each bit can be set individually as either an input or output by P7DIR register. A pull-up resistor for each bit can be selected individually by P7PLU register. At reset, the input mode is selected and pull-up resistor is disabled (high impedance).
P71	50	44			
P72	49	43			
P73	48	42			
P74	47	41			
P75	46	40			
P76	45	39			
P77	44	38			
P80	43	37	I/O	I/O port 8	8-bit CMOS tri-state I/O port. Each bit can be set individually as either an input or output by P8DIR register. A pull-up resistor for each bit can be selected individually by P8PLU register. At reset, the input mode is selected and pull-up resistor is disabled (high impedance).
P81	42	36			
P82	41	35			
P83	40	34			
P84	39	33			
P85	38	32			
P86	37	31			
P87	36	30			
P90	12	8	I/O	I/O port 9	5-bit CMOS tri-state I/O port. Each bit can be set individually as either an input or output by P9DIR register. A pull-up /pull-down resistor for each bit can be selected individually by P9PLUD register. A pull-up/down resistor connection for each port can be selected individually in SELUD register. A pull-up/pull down can not be mixed. At reset, the input mode is selected and pull-up resistor is disabled (high impedance).
P91	13	9			
P92	74	-			
P93	75	-			
P94	76	60			
PA0	1	61	I/O	I/O port A	8-bit CMOS tri-state I/O port. Each bit can be set individually as either an input or output by PADIR register. A pull-up resistor for each bit can be selected individually by PAPLU register. At reset, the input mode is selected and pull-up resistor is disabled (high impedance).
PA1	2	62			
PA2	3	63			
PA3	4	64			
PA4	5	1			
PA5	6	2			
PA6	7	3			
PA7	8	4			

Table remarks -: Without function

Pins	MN101EF A8/A3 Series	MN101EF A7/A2 Series	I/O	Function	Description
PB0	80	-	I/O	I/O port B	4-bit CMOS tri-state I/O port. Each bit can be set individually as either an input or output by PBDIR register. A pull-up /pull-down resistor for each bit can be selected individually by PBPLUD register. A pull-up/down resistor connection for each port can be selected individually in SELUD register. A pull-up/pull down can not be mixed. At reset, the input mode is selected and pull-up resistor is disabled (high impedance).
PB1	79	-			
PB2	78	-			
PB3	77	-			
SBO0A	26	21	Output	Serial interface transmission data output pins	Transmission data output pins for serial interface 0,1,2,4. The output configuration, either COMS push-pull or Nch open-drain can be selected in P0ODC, P3ODC, P4ODC, P5ODC, P6ODC and P7ODC registers. Pull-up resistor can be selected in P0PLU, P3PLUD, P4PLU, P5PLU(D), P6PLU, and P7PLU registers. Select output mode in P0DIR, P3DIR, P4DIR, P5DIR, P6DIR, and P7DIR registers and set serial data output mode in serial mode register 1 (SC0MD1, SC1MD1, SC2MD1, SC4MD1). These can be used as normal I/O pins when serial interface is not used.
SBO0B	70	-			
SBO1A	58	52			
SBO1B	46	40			
SBO2	54	48			
SBO4A	50	44			
SBO4B	72	-			
SBI0A	25	20	Input	Serial interface reception data input pins	Reception data input pins for serial interface 0,1,2,4. Pull-up resistor can be selected in P0PLU, P3PLUD, P4PLU, P5PLU(D), P6PLU and P7PLU registers. Select the output mode in P0DIR, P3DIR, P4DIR, P5DIR, P6DIR and P7DIR registers and select serial data input mode in serial mode register 1 (SC0MD1, SC1MD1, SC2MD1, SC4MD1). These can be used as normal I/O pins when serial interface is not used.
SBI0B	69	-			
SBI1A	59	53			
SBI1B	45	39			
SBI2	53	47			
SBI4A	51	45			
SBI4B	71	-			
SBT0A	27	22	I/O	Serial interface Clock I/O pins	Clock I/O pins for serial interface 0,1,2,4. The output configuration, either COMS push-pull or Nch open-drain can be selected in P0ODC, P3ODC, P4ODC, P5ODC, P6ODC and P7ODC registers. Pull-up resistor can be selected in P0PLU, P3PLUD, P4PLU, P5PLU(D), P6PLU and P7PLU registers. Select clock I/O in P0DIR, P3DIR, P4DIR, P5DIR, P6DIR and P7DIR registers and serial mode register 1 (SC0MD1, SC1MD1, SC2MD1, SC4MD1) with the communication mode. These can be used as normal I/O pins when serial interface is not used.
SBT0B	68	-			
SBT1A	60	54			
SBT1B	44	38			
SBT2	52	46			
SBT4A	49	43			
SBT4B	72	-			
TXD0A	26	21	Output	UART transmission data output pins	In serial interface 0,1,2 in UART mode, this pin is configured as the transmission data output pin. The output configuration, either COMS push-pull or Nch open-drain can be selected in P0ODC, P4ODC, P5ODC, P6ODC and P7ODC registers. Pull-up resistor can be selected by P0PLU, P4PLU, P5PLU(D), P6PLU and P7PLU registers. Select the output mode in P0DIR, P4DIR, P5DIR, P6DIR and P7DIR registers and select serial data output mode in serial mode register 1 (SC0MD1, SC1MD1, SC2MD1). These can be used as normal I/O pins when serial interface is not used.
TXD0B	70	-			
TXD1A	58	52			
TXD1B	46	40			
TXD2	54	48			
RXD0A	25	20	Input	UART reception data output pins	In serial interface 0,1,2 in UART mode, this pin is configured as the reception data input pin. Pull-up resistor can be selected in P0PLU, P4PLU, P5PLU(D), P6PLU and P7PLU registers. Select the input mode in P0DIR, P4DIR, P5DIR, P6DIR and P7DIR registers and select serial input in serial mode register 1 (SC0MD1, SC1MD1, SC2MD1). These can be used as normal I/O pins when serial interface is not used.
RXD0B	69	-			
RXD1A	59	53			
RXD1B	45	39			
RXD2	53	47			
SDA4A	50	44	I/O	IIC data I/O pins	In serial interface 4 in IIC mode, this pin is configured as the data I/O pin. For the output configuration, select Nch open-drain in P3ODC and P7ODC register and set pull-up resistor in P3PLUD and P7PLU register. Select the output mode in P0DIR register and select serial data I/O mode by serial mode register 1 (SC4MD1). These can be used as normal I/O pin when serial interface is not used.
SDA4B	72	-			
SCL4A	49	43	I/O	IIC clock I/O pins	In serial interface 4 in IIC mode, this pin is configured as the clock I/O pin. For the output configuration, select Nch open-drain in P0ODC and P7ODC register and set pull-up resistor by P0PLU and P7PLU register. Select the output mode at P0DIR register and select serial clock I/O mode in serial mode register 1 (SC4MD1). These can be used as normal I/O pin when serial interface is not used
SCL4B	72	-			

Table remarks -: Without function

Pins	MN101EF A8/A3 Series	MN101EF A7/A2 Series	I/O	Function	Description
TM0IOA	1	61	I/O	Timer I/O pins	Event counter clock input pin, timer output and PWM signal output pin for 8-bit timer 0 to 3. To use this pin as event clock input, configure it as input by P0DIR, P6DIR and PADIR register. In the input mode, pull-up resistor can be selected in P0PLU, P6PLU, and PAPLU registers. For timer output, PWM signal output, select the special function pin in P0OMD1, P0OMD2, P6OMD and PAOMD registers, and set to the output mode in P0DIR, P6DIR and PADIR registers. These can be used as normal I/O pins when Timer I/O pin is not used.
TM0IOB	27	22			
TM1IOA	2	62			
TM1IOB	57	51			
TM2IOA	3	63			
TM2IOB	27	22			
TM3IOA	4	64			
TM3IOB	56	50			
BUZZERA	65	59	Output	Buzzer output pins	Piezoelectric buzzer driving pin. Buzzer output is available to Port 5, 8. The driving frequency can be set in DLYCTR register. In order to select Buzzer output, select the special function pin in P5OMD, P8OMD register, and set P5DIR, P8DIR register to the output mode. At the same time, select Buzzer output in oscillation stabilization wait control register (DLYCTR). These can be used as normal I/O pins when Buzzer output is not used.
BUZZERB	36	30			
NBUZZERA	64	58			
NBUZZERB	37	31			
TM7IOA	6	2	I/O	Timer I/O pins	Event counter clock input pin, timer output and PWM signal output pin for 16-bit timer7,8 and 9. To use this pin as event clock input, configure it as input with P0DIR and PADIR registers. In the input mode, pull-up resistor can be selected by P0PLU and PAPLU registers. For timer output, PWM signal output, select the special function pin in P0OMD1 and PAOMD registers, and set to the output mode in P0DIR and PADIR registers. These can be used as normal I/O pins when not used as timer I/O pins.
TM7IOB	25	20			
TM8IOA	7	3			
TM8IOB	26	21			
TM9IOA	8	4			
TM9IOB	24	19			
TM9OD0	43	37	Output	Timer PWM output	PWM signal output pin for 16-bit timer 9. Select the special function pin in P8OMD register, and set to the output mode in P8DIR register. These can be used as normal I/O pins when not used as timer I/O pins.
TM9OD1	42	36			
TM9OD2	41	35			
TM9OD3	40	34			
TM9OD4	39	33			
TM9OD5	38	32			
VREF+	9	5	-	A/D reference voltage input pin	Reference power supply pin for A/D converter. Normally, the values of $V_{REF+} = V_{DD5}$ is used.
AN0	1	61	input	Analog input pins	[MN101EFA8/A3 Series] Analog input pins for 16-channel, 10-bit A/D converter. Select the analog input by P3IMD, P9IMD, PAIMD, PBIMD register. When not used for analog input, these pins can be used as normal input pins. [MN101EFA7/A2 Series] Analog input pins for 12-channel, 10-bit A/D converter. Select the analog input by P5IMD, P9IMD, PAIMD register. When not used for analog input, these pins can be used as normal input pins.
AN1	2	62			
AN2	3	63			
AN3	4	64			
AN4	5	1			
AN5	6	2			
AN6	7	3			
AN7	8	4			
AN8	80	60			
AN9	79	59			
AN10	78	58			
AN11	77	57			
AN12	76	-			
AN13	75	-			
AN14	74	-			
AN15	73	-			

Table remarks -: Without function

Pins	MN101EF A8/A3 Series	MN101EF A7/A2 Series	I/O	Function	Description
IRQ0	31	25	Input	External interrupt	External interrupt input pins. Select the external interrupt input enable by IRQCNT register. The valid edge for IRQ0 to 4 can be selected with IRQnICR register. IRQ2 to 4 can be set at both edges at pin voltage level. When not used for interrupts, these can be used as normal input pins.
IRQ1	32	26			
IRQ2	33	27			
IRQ3	34	28			
IRQ4	35	29			
KEY0	51	45	Input	Key interrupt input pins	Input pins for KEY interrupt based on OR condition result of pin inputs. These can be set to key input pins by 1-bit with KEY interrupt control register (KEYT3_1IMD, KEY3_2_IMD). When not used for KEY input, these pins can be used as normal I/O pins.
KEY1	50	44			
KEY2	49	43			
KEY3	48	42			
KEY4	47	41			
KEY5	46	40			
KEY6	45	39			
KEY7	44	38			
LED0	1	61	Output	LED drive pins	Large current output pins. Select the large current output by LEDCNT registers. When not used for LED output, these pins can be used as normal I/O pins.
LED1	2	62			
LED2	3	63			
LED3	4	64			
LED4	5	1			
LED5	6	2			
LED6	7	3			
LED7	8	4			
DMOD	20	16	Input	Mode switch input pins	Set always to V_{DD5} level.
MMOD	18	14	Input	ROM area switch input pins at start	Set always to V_{SS} level.
TS0IN0	80	60	Input	Touch sensor input pins These pins are not equipped in MN101EFA3 Series and MN101EFA2 Series.	Input pins for Touch Sensor Timer of 12 channels (8 channels of MN101EFA7 Series). Set "Used" to corresponding channel by TS0TCHSEL, TS1TCHSEL register. This setup is available regardless of the setting of port control registers. These can be used as normal I/O pins when Touch Sensor Timer is not used.
TS0IN1	79	59			
TS0IN2	78	58			
TS0IN3	77	57			
TS0IN4	76	56			
TS0IN5	75	55			
TS0IN6	74	54			
TS0IN7	73	53			
TS1IN0	68	-			
TS1IN1	67	-			
TS1IN2	66	-			
TS1IN3	65	-			
TS0RC	72	52	Input	Touch sensor resistor connect pins. These pins are not equipped in MN101EFA3 Series and MN101EFA2 Series.	These are used in the following cases. 1. External resistor connection for Touch Sensor Timer Set both TS0MD, TS1MD and RS0MD, RS1MD of TS0TMD, TS1TMD register to "1". 2. The capacitor/resistor connection for Touch Sensor using A/D converter Set TS0ADCNT, TS1ADCNT register. This setup is available regardless of the setting of port control registers. These can be used as normal I/O pins when Touch Sensor Timer is not used.
TS1RC	69	-			
TS0OP	71	51	Output		
TS1OP	70	-			

For the MMOD setup in rewriting the flash memory, refer to [Chapter 16 16.5 User Mode Microcontroller Rewriting], [Chapter 16 16.6 BOOT Mode Microcontroller Rewriting], [Chapter 16 16.7 Appendix].

1.4 Block Diagram

1.4.1 Block Diagram

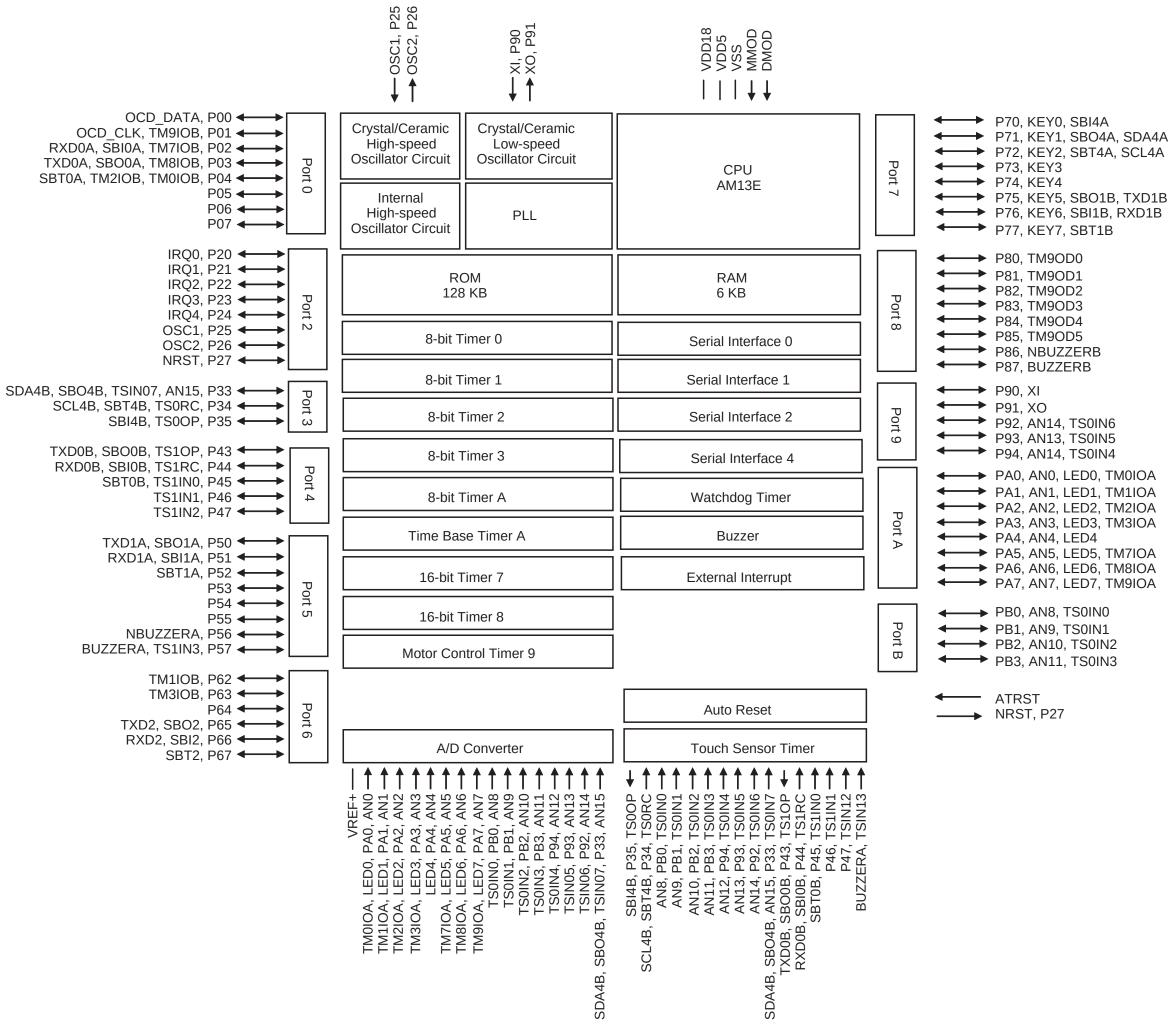


Figure:1.4.1 Block Diagram

* Varies depending on models.
Refer to [Chapter 1 1.1.2 Product Summary] and [Chapter 1 1.3.3 Pin Functions].

1.5 Electrical Characteristics

This LSI manual describes standard specifications.

When using this LSI, consult our sales offices for the product specifications.

Structure	CMOS integrated circuit
Application	General-purpose
Function	CMOS 8-bit single chip microcomputer

1.5.1 Absolute Maximum Ratings

A. Absolute Maximum Ratings *2 *3 *4 V_{SS} = 0 V

Parameter			Symbol	Rating	Unit
A1	Power supply voltage		V_{DD5}	-0.3 to +7.0	V
A2	Power supply voltage		V_{DD18}	-0.3 to +2.5	
A3	Input pin voltage		V_I	-0.3 to V_{DD5} +0.3 (upper limit: 7.0 V)	
A4	Output pin voltage		V_O	-0.3 to V_{DD5} +0.3 (upper limit: 7.0 V)	
A5	I/O pin voltage		V_{IO1}	-0.3 to V_{DD5} +0.3 (upper limit: 7.0 V)	
A6	Peak output current	LED output	I_{OL1} (peak)	30	mA
A7		Other than LED output	I_{OL2} (peak)	20	
A8		All pins	I_{OH} (peak)	-10	
A9	Average output current *1	LED output	I_{OL1} (avg)	20	
A10		Other than LED output	I_{OL2} (avg)	15	
A11		All pins	I_{OH} (avg)	-5	
A12	Power dissipation		P_D	400	mW
A13					
A14					
A15					
A16	Operating ambient temperature		T_{opr}	-40 to +85	°C
A17	Storage temperature		T_{STG}	-55 to +125	

*1 Applied to any 100 ms period.

*2 Connect at least one bypass capacitor of 0.1 µF + 1.0 µF or larger between VDD5 pin and GND for the internal power voltage stabilization.

*3 Connect appropriate capacitor about 0.1 µF + 1.0 µF between VDD18 pin and VSS pin, near the microcontroller according to the Figure:1.5.1 shown below for the internal power supply stabilization.

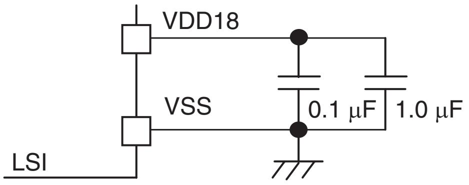


Figure:1.5.1 Capacitor Connection between VDD18 and VSS Pins

*4 The absolute maximum ratings are the limit values beyond which the LSI may be damaged.

1.5.2 Operating Conditions

B. Operating Conditions

V_{SS} = 0 V
Ta = -40 °C to +85 °C

Parameter		Symbol	Conditions	Rating			Unit
				MIN	TYP	MAX	
Power supply voltage *5							
B1	Power supply voltage	V_{DD1}		4.0		5.5	V
B2	RAM retention power supply voltage	V_{DD2}	During STOP mode	2.2		5.5	
Operating speed *6							
B3	Instruction execution time fs	t_{c1}	V_{DD5} = 4.0 V to 5.5 V (When ROMHND of HANDSHAKE register is “1”.)	0.05			μs
B4		t_{c2}	V_{DD5} = 4.0 V to 5.5 V (When ROMHND of HANDSHAKE register is “0”.)	0.10			
B5		t_{c3}	V_{DD5} = 4.0 V to 5.5 V	61			

*5 fs: Machine clock frequency

*6 tc1 to 2 : when the machine clock is selected from external high-speed oscillation, internal high-speed oscillation, or both the oscillations multiplied by PLL.

tc7: when the machine clock is selected from external low-speed oscillation.

External Oscillator 1 Figure:1.5.2

B6	Frequency	f_{hosc1}	V_{DD5} is within the specified operating power supply voltage range. (Refer to the ratings of B1 to B2 for the operating supply voltage range)	2.0		10	MHz
B7	Internal feedback resistor	R_{f10}	V_{DD5} = 5.0 V		980		kΩ

External Oscillator 2 Figure:1.5.2

B8	Frequency	f_{sosc1}	V_{DD5} = 4.0 V to 5.5 V		32.768		kHz
B9	Internal feedback resistor	R_{f20}	V_{DD5} = 5.0 V		6.2		MΩ

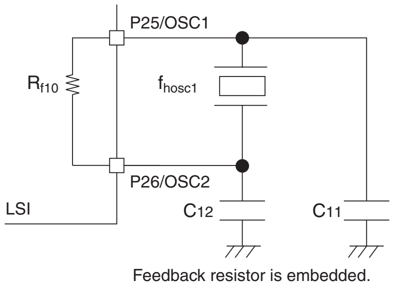


Figure:1.5.2 External Oscillator 1

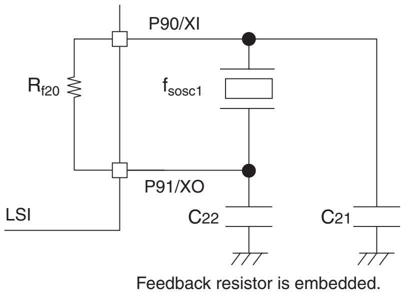


Figure:1.5.3 External Oscillator 2

Connect external capacitors suited for the used oscillator.
The reference value denotes external capacity value based on our matching result.
When crystal oscillator or ceramic oscillator is used, the oscillation frequency is changed depending on the value of capacitor. For external capacity value, please consult the oscillator manufacturer and perform matching tests enough for determining appropriate values.

V_{DD5} = 4.0 V to 5.5 V
V_{SS} = 0 V
Ta = -40 °C to +85 °C

Parameter		Symbol	Conditions	Rating			Unit
				MIN	TYP	MAX	
External clock input 1 OSC1 (OSC2 is unconnected)							
B10	Clock frequency	f_{hosc2}		2		10.0	MHz
B11	High-level pulse width *7	t_{wh1}	Figure:1.5.4	45			ns
B12	Low-level pulse width *7	t_{wl1}		45			
B13	Rising time	t_{wr1}	Figure:1.5.4	0		5.0	
B14	Falling time	t_{wf1}		0		5.0	

*7 The clock duty ratio should be 45 % to 55 %

External clock input 2 XI (XO is unconnected)

B15	Clock frequency	f_{sosc2}			32.768		kHz
B16	High-level pulse width *7	t_{wh2}	Figure:1.5.5		4.5		µs
B17	Low-level pulse width *7	t_{wl2}			4.5		µs
B18	Rising time	t_{wr2}	Figure:1.5.5	0		20	ns
B19	Falling time	t_{wf2}		0		20	ns

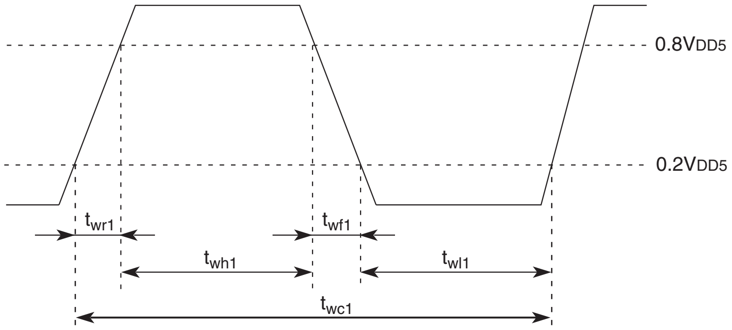


Figure:1.5.4 OSC1 Timing Chart

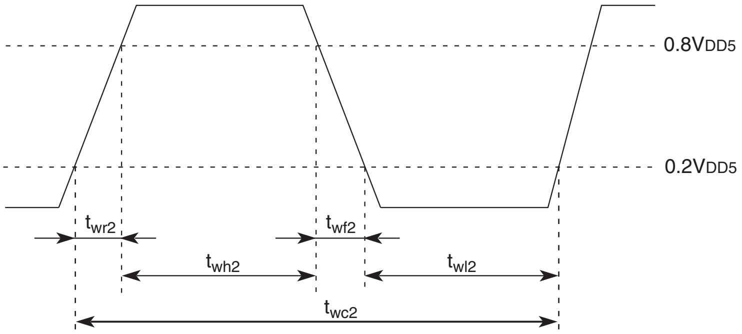


Figure:1.5.5 XI Timing Chart

1.5.3 DC Characteristics

C. DC Characteristics

V_{SS} = 0 V
Ta = -40 °C to +85 °C

	Parameter	Symbol	Conditions	Rating			Unit
				MIN	TYP	MAX	
Power supply current *8							
C1	Power supply current during operation	I_{DD1}	V_{DD5}=5 V fosc=10 MHz [Double-speed mode: fs=fosc] (PLL is not used) *9		5	14	mA
C2		I_{DD2}	V_{DD5}=5 V fosc=10 MHz [Multiplied by 2, Divided by 2: fs=fosc] (PLL is used) *9		6	18	
C3		I_{DD3}	V_{DD5}=5 V fosc=10 MHz [Multiplied by 2: fs=20 MHz] (PLL is used) *9		9	20	
C4		I_{DD4}	V_{DD5}=5 V frc=16 MHz [Double-speed mode: fs=16 MHz] (PLL is not used) *9		6	15	
C5	Power supply current during operation	I_{DD5}	V_{DD5}=5 V fx=32.768 kHz [fs=fx/2]		200	400	µA
C6	Power supply current during STOP mode	I_{DD6}	V_{DD5}=5 V		145	245	µA

[*8] Measured without loading (pull-up and pull-down resistors are not connected.)

To measure the power supply current during operation I_{DD1} to I_{DD4};
1. Set all I/O pins to input mode,
2. Set the CPU mode to “NORMAL mode”,
3. Fix pin MMOD to V_{SS} level and input pins to V_{DD5} level
4. Input the rectangular wave of 10 MHz with amplitude of V_{DD5} and V_{SS}, from pin OSC1.

To measure the power supply current during SLOW mode I_{DD5};
1. Set all I/O pins to input mode
2. Set the CPU mode to "SLOW mode"
3. Fix the MMOD to V_{SS} level and input pins to V_{DD5} level

To measure the power supply current during STOP mode I_{DD6};
1. Set the CPU mode to “STOP mode”,
2. Fix pin MMOD to V_{SS} level and input pin to V_{DD5} level
3. Open pin OSC1.

[*9] When ROMHND of HANDSHAKE register is set to “1”

V_{DD5} = 4.0 V to 5.5 V V_{SS} = 0 V
Ta = -40 °C to +85 °C

Parameter		Symbol	Conditions	Rating			Unit
				MIN	TYP	MAX	
Input pin 1 ATRST, MMOD							
C7	Input high voltage	V_{IH1}		0.8V_{DD5}		V_{DD5}	V
C8	Input low voltage	V_{IL1}		0		0.2V_{DD5}	
C9	Input leakage current	I_{LK1}	V_{IN} = 0 V to V_{DD5}			± 2	μA
Input pin 2 P27/NRST							
C10	Input high voltage	V_{IH2}		0.8V_{DD5}		V_{DD5}	V
C11	Input low voltage	V_{IL2}		0		0.15V_{DD5}	
C12	Pull-up resistor	R_{RH2}	V_{DD5}=5 V, V_{IN}= V_{SS}	10	50	100	kΩ

V_{DD5} = 4.0 V to 5.5 V V_{SS} = 0 V
Ta = -40 °C to +85 °C

Parameter		Symbol	Conditions	Rating			Unit
				MIN	TYP	MAX	

Input pin 3
P00 to P07, P20 to P26, P43 to P47, P50 to P57, P62 to P67, P70 to P77, P80 to P87
(MN101EFA8/A3 Series)
P00 to P06, P20 to P26, P62 to P67, P70 to P77, P80 to P87 (MN101EFA7/A2 Series)

C13	Input high voltage	V_{IH3}		$0.8V_{DD5}$		V_{DD5}	V
C14	Input low voltage	V_{IL3}		0		$0.2V_{DD5}$	
C15	Input leakage current	I_{LK3}	V_{IN}=0 V to V_{DD5}			± 2	µA
C16	Pull-up resistor	R_{RH3}	V_{DD5}=5.0 V, V_{IN}=V_{SS} Pull-up resistor ON	10	50	100	kΩ
C17	Output high voltage	V_{OH3}	V_{DD5}=5.0 V, I_{OH}=-0.5 mA	4.5			V
C18	Output low voltage	V_{OL3}	V_{DD5}=5.0 V, I_{OL}=1.0 mA			0.5	

Input pin 4 PA0 to PA7

C19	Input high voltage	V_{IH4}		$0.8V_{DD5}$		V_{DD5}	V
C20	Input low voltage	V_{IL4}		0		$0.2V_{DD5}$	
C21	Input leakage current	I_{LK4}	V_{IN}=0 V to V_{DD5}			± 2	µA
C22	Pull-up resistor	R_{RH4}	V_{DD5}=5.0 V, V_{IN}=V_{SS} Pull-up resistor ON	10	50	100	kΩ
C23	Output high voltage	V_{OH4}	V_{DD5}=5.0 V, I_{OH}=-0.5 mA	4.5			V
C24	Output low voltage 1	V_{OL41}	V_{DD5}=5.0 V, I_{OL}=1.0 mA LED output OFF			0.5	
C25	Output low voltage 2	V_{OL42}	V_{DD5}=5.0 V, IOL=15.0 mA LED output ON			1.0	

Input pin 5
P33 to P35, P90 to P94, PB0 to PB3 (MN101EFA8/A3 Series)
P50 to P57, P90, P91, P94 (MN101EFA7/A2 Series)

C26	Input high voltage	V_{IH5}		$0.8V_{DD5}$		V_{DD5}	V
C27	Input low voltage	V_{IL5}		0		$0.2V_{DD5}$	
C28	Input leakage current	I_{LK5}	V_{IN}=0 V to V_{DD5}			± 2	µA
C29	Pull-up resistor	R_{RH5}	V_{DD5}=5.0 V, V_{IN}=V_{SS} Pull-up resistor ON	10	50	100	kΩ
C30	Pull-down resistor	R_{RL5}	V_{DD5}=5.0 V, V_{IN}=V_{DD5} Pull-down resistor ON	10	50	100	
C31	Output high voltage	V_{OH5}	V_{DD5}=5.0 V, I_{OH}=-0.5 mA	4.5			V
C32	Output low voltage	V_{OL5}	V_{DD5}=5.0 V, I_{OL}=1.0 mA			0.5	

V_{DD5} = 4.0 V to 5.5 V V_{SS} = 0 V
Ta = -40 °C to +85 °C

Parameter		Symbol	Conditions	Rating			Unit
				MIN	TYP	MAX	
Input pin 6 DMOD							
C33	Input high voltage	V_{IH6}		0.8V_{DD5}		V_{DD5}	V
C34	Input low voltage	V_{IL6}		0		0.2V_{DD5}	
C35	Pull-up resistor	R_{RH6}	V_{DD5}=5.0 V, V_{IN}=V_{SS} Pull-up resistor ON	10	50	100	kΩ

1.5.4 A/D Converter Characteristics

D. A/D Converter Characteristics *11

V_{DD5} = 5.0 V V_{SS} = 0 V
Ta = -40 °C to +85 °C

Parameter		Symbol	Conditions	Rating			Unit
				MIN	TYP	MAX	
D1	Resolution					10	Bits
D2	Non-linearity error 1		V_{DD5}=5.0 V, V_{SS}=0 V V_{REF+}=5.0 V T_{AD}=800 ns			± 3	LSB
D3	Differential non-linearity error 1					± 3	
D4	Zero transition voltage		V_{DD5}=5.0 V, V_{SS}=0 V V_{REF+}=5.0 V T_{AD}=800 ns		10	30	mV
D5	Full-scale transition voltage			4970	4990		
D6	A/D conversion time		T_{AD}=800 ns	12.93			μs
D7	Sampling time		T_{AD}=800 ns	1.6			
D8	Reference voltage	V_{REF+}	Note)	4.0		V_{DD5}	V
D9	Analog input voltage			V_{SS}		V_{REF+}	
D10	Analog input leakage current		Channel OFF V_{ADIN}=V_{SS} to V_{DD5}			± 2	μA
D11	Reference voltage pin input leakage current		Ladder resistance OFF $V_{SS} \leq V_{REF+} \leq V_{DD5}$			± 5	
D12	Ladder resistance	R_{LADD}	V_{DD5}=5.0 V	15	40	80	kΩ

*11 T_{AD} is A/D conversion clock cycle.
The specification values of D2 to D5 are guaranteed on the condition of V_{DD5}=V_{REF+}=5 V, V_{SS}=0 V.

Even if A/D function is not used, the voltage of VREF+ pin must be set between V_{DD5} and 4.0 V.

1.5.5 Auto Reset Characteristics

E. Auto Reset Characteristics

V_{DD5} = V_{RST} to 5.5 V V_{SS} = 0 V
Ta = -40 °C to +85 °C

Parameter		Symbol	Conditions	Rating			Unit
				MIN	TYP	MAX	
Power supply voltage							
E1	Operating supply voltage	V_{DD7}	Auto reset is used	V_{RST}		5.5	V
Power supply voltage							
E2	Power detection level	V_{RST1}	At rising	4.10	4.30	4.50	V
E3	Power detection level	V_{RST2}	At falling	4.00	4.20	4.40	
E4	Supply voltage change rate	Δt/ΔV		2			ms/V

1.5.6 Internal High-speed Oscillation Circuit

F. Internal High-speed Oscillation Circuit

V_{DD5} = 4.0 V to 5.5 V V_{SS} = 0 V

Parameter		Symbol	Conditions	Rating			Unit
				MIN	TYP	MAX	
F1	Internal high-speed oscillation circuit frequency	f_{rc}	Ta = -40 °C to +85 °C		16		MHz
F2	Temperature dependence of oscillation frequency	f_{rc3}	Ta = 25 °C	-5.0		5.0	%
F3		f_{rc4}	Ta = -40 °C to +85 °C				

1.5.7 Flash EEPROM Program Conditions

G. Flash EEPROM Program Conditions

V_{DD5} = 4.0 V to 5.5 V V_{SS} = 0 V
Ta = -40 °C to +85 °C

Parameter		Symbol	Conditions	Rating			Unit
				MIN	TYP	MAX	
G1	Programming/Erasing times of 32KB, 20KB Sector *2	E_{MAX1}		1000			Times
G2	Programming/Erasing times of 4KB Sector *2	E_{MAX2}		10000			Times
G3	Data retention period of 32KB, 20KB Sector *1	T_{HOLD1}	Ta= 85°C, P/E times ≤ 1000	20			Years
G4	Data retention period of 4KB Sector *1	T_{HOLD2}	Ta= 85°C, P/E times ≤ 1000 *2	20			Years
		T_{HOLD3}	Ta= 65°C, P/E times ≤ 10000 *2	20			Years

*1 Contain the period when power supply voltage is not supplied.

*2 Programming/Erasing times(P/E Times) is counted by the number of time a sector is erased.
It is controlled on sector basis.
For example, if writing 1 byte of data in any sector for hundred of times and then erasing the sector, a single rewriting is counted. Also, the number of times of rewriting in another sector, in which erasing is not performed, is not counted.
Overwriting data is disabled. To rewrite data, write the data after erasing sectors.

1.6 Package Dimension

- Package code: TQFP080-P-1212F Unit: mm

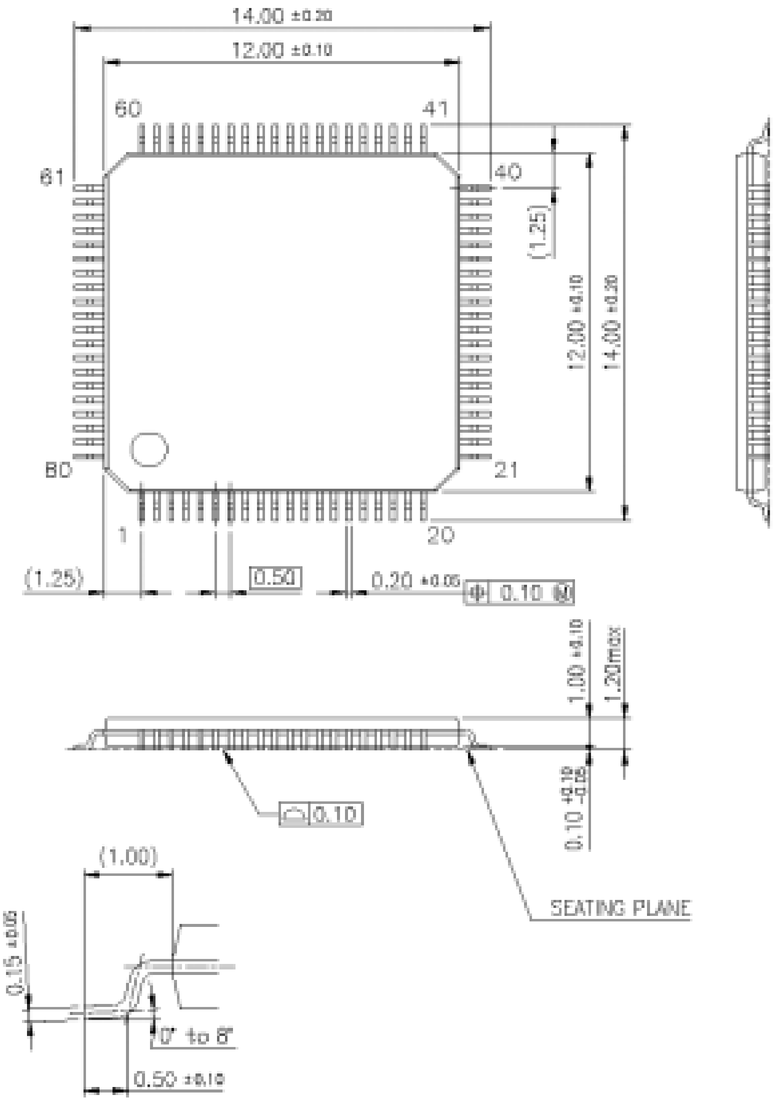


Figure:1.6.1 80-pin TQFP Package Dimension

This package dimension is subject to change. Before using this product, please obtain product specifications from our sales offices.

■ Package code: LQFP080-P-1414E Unit: mm

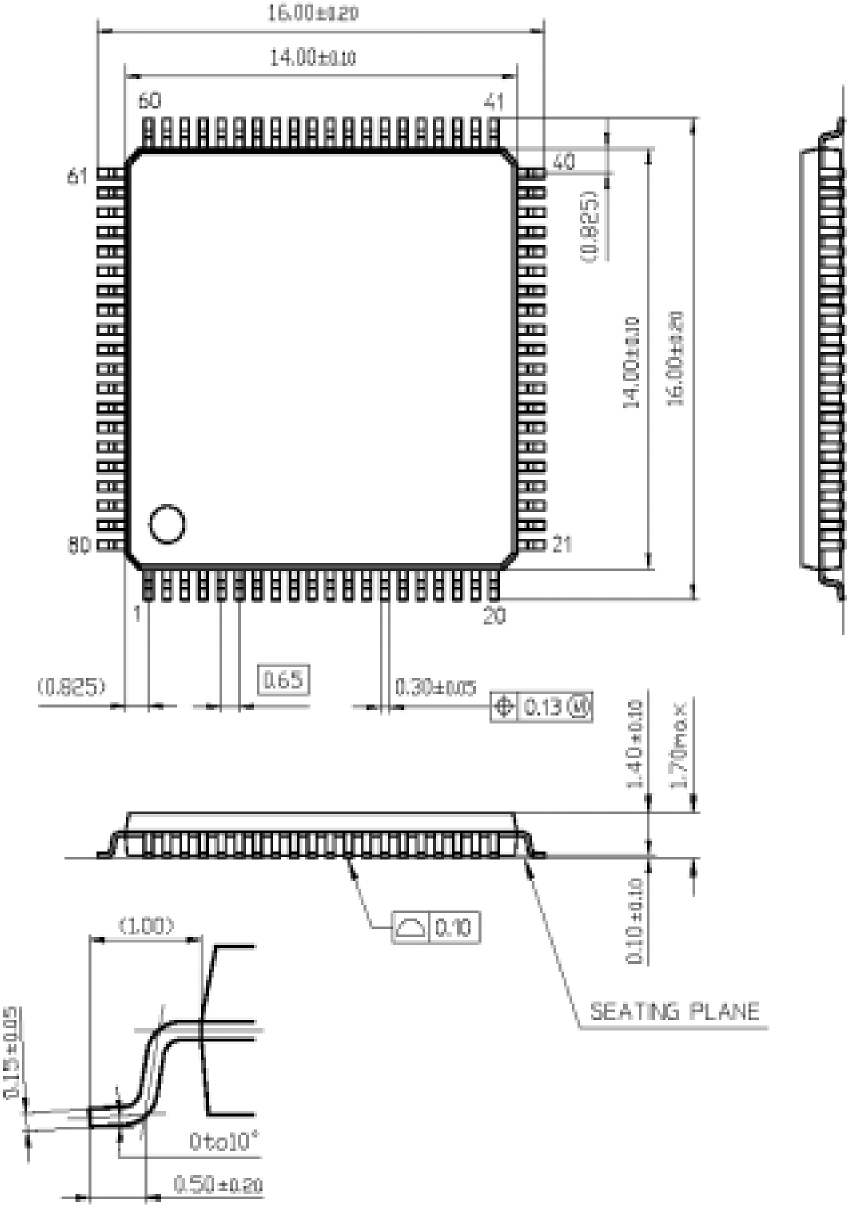


Figure:1.6.2 80-pin LQFP Package Dimension

This package dimension is subject to change. Before using this product, please obtain product specifications from our sales offices.

- Package code: TQFP064-P-1010D Unit: mm

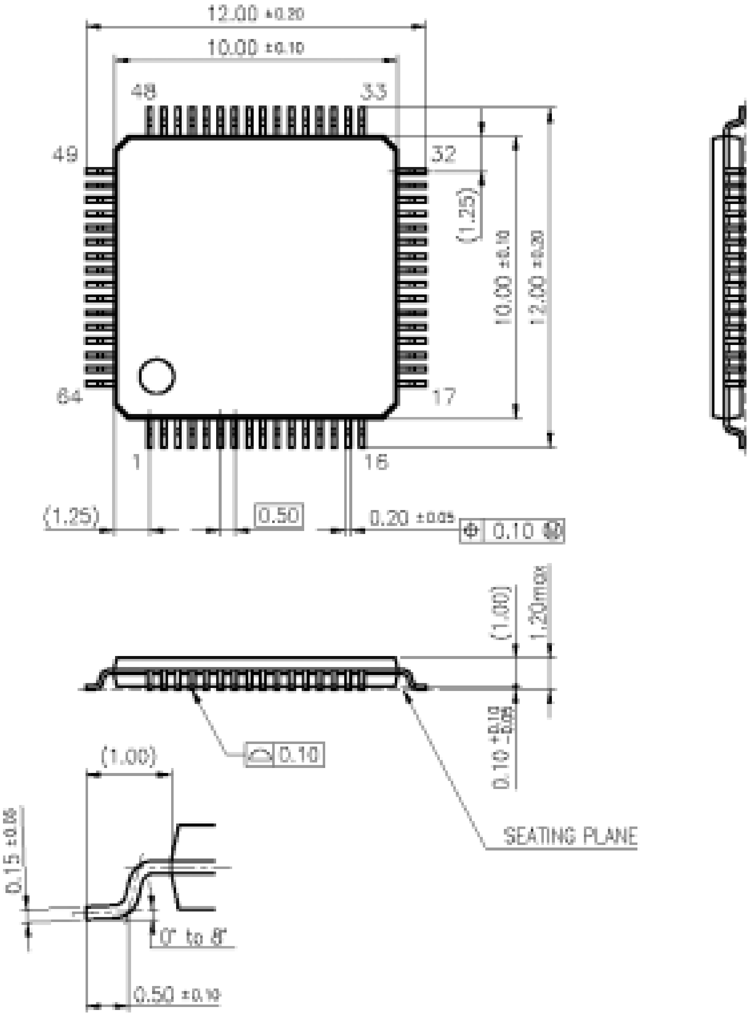


Figure:1.6.3 64-pin TQFP Package Dimension

This package dimension is subject to change. Before using this product, please obtain product specifications from our sales offices.

■ Package code: LQFP064-P-1414 Unit: mm

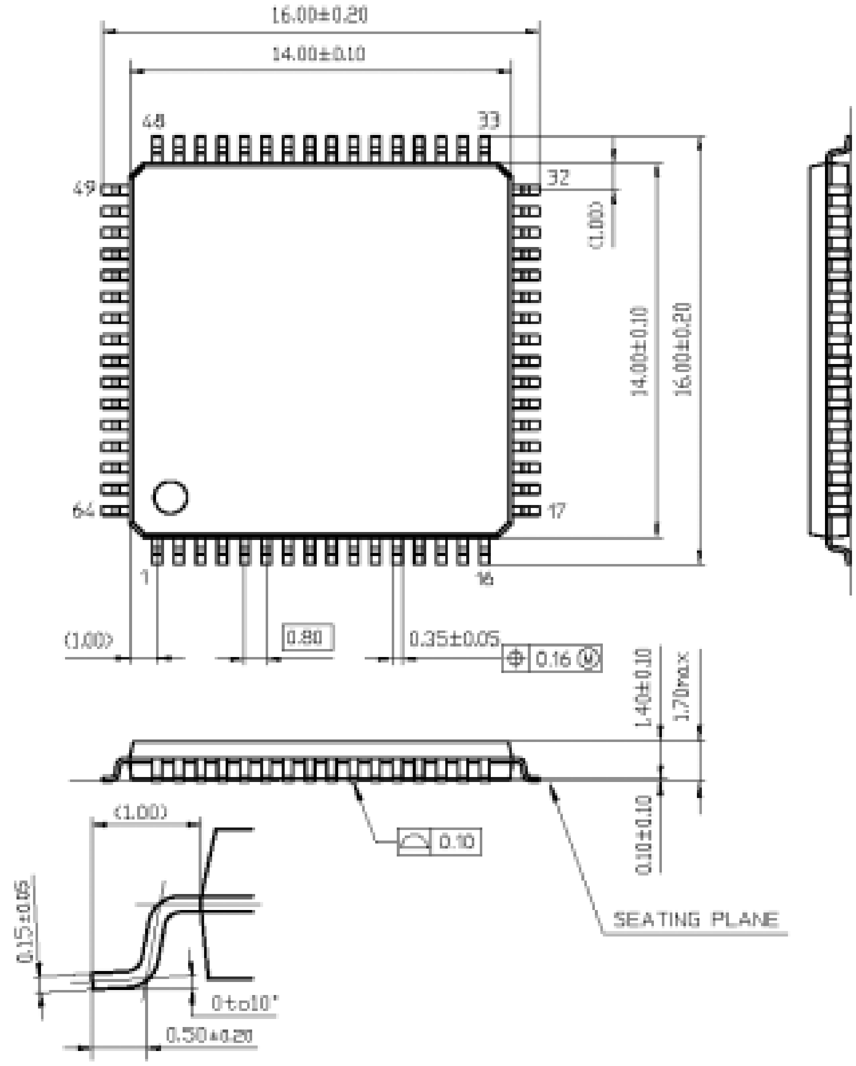


Figure:1.6.4 64-pin LQFP Package Dimension

This package dimension is subject to change. Before using this product, please obtain product specifications from our sales offices.

1.7 Cautions for Circuit Setup

1.7.1 General Usage

■ Connection of VDD5 pin and VSS pin

All of VDD5 and VSS pins should be connected directly to the power source and ground in the external. Put them on printed circuit board after the location of LSI (package) pin is confirmed. Connection error may lead a fusion and breakdown of a microcontroller.

■ VREF+ pin Connection

When using VREF+ pin and VDD5 pin in the same potential, separate VREF+ pin at the root of the power supply.

■ Cautions for Operation

1. If you install the product close to high-field emissions (under the cathode ray tube, etc.), shield the package surface to ensure normal performance.
2. Operation temperature should be well considered. Each product has different condition. For example, if the operation temperature is over the condition, improper operation could be occurred.
3. Operation voltage should be also well considered. Each product has different operating range.
    - If the operation voltage is over the operating range, duration of the product could be shortened.
    - If the operation voltage is below the operating range, improper operation could be occurred.

1.7.2 Oscillator

This LSI’s oscillation clock can be used with a ceramic and crystal oscillator.

■ Recommended oscillators

Figure:1.7.1 show basic configuration connected with a ceramic oscillator, and Table:1.7.1 shows recommended oscillators and the circuit constants.

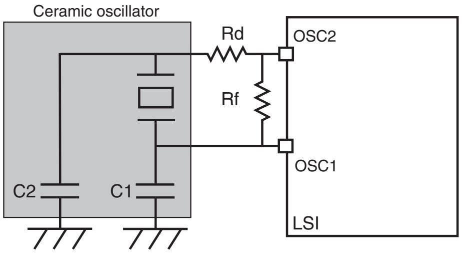


Figure:1.7.1 Basic Configuration of Oscillator Connection

Table:1.7.1 Recommended oscillators and the circuit constants

Product name (Series)	Manufacturer	Frequency [Hz]	Type	Oscillator Product Number	Recommended circuit constant				Recommended operation condition	
					C1[pF]	C2[pF]	Dumping resistor value Rd[Ω]	Feedback resistor value Rf[Ω]	Power supply voltage[V]	Tempera-ture range [°C]
MN101EFA8 MN101EFA7 MN101EFA3 MN101EFA2	Murata Manufacturing Company, Ltd.	2.000M	SMD	Under Evaluation						-40 to 85
		4.000M	SMD							
			Read							
		8.000M	SMD							
			Read							
		10.000M	SMD							
			Read							

Note): ( ) denotes internal capacity
The end of product name, after "-", denotes package specification.
-SMD type [-R0: plastic taping (ϕ=180mm), -B0: Individual]
-Read type[-A0: Flat pack (Ho=18mm), -B0: Individual

The above recommended value is the result of oscillator evaluation only on this LSI. After an evaluation on a set board, insert dumping resistor if needed.
Crystal oscillator is not evaluated. So consult the oscillator manufacturer for the appropriate circuit constants.

Circuit constant of each ceramic or crystal oscillator, which is connected to OSC1/OSC2, differs depending on stray capacitance of the oscillator or on the mounting circuit. So consult the oscillator manufacturer for the appropriate circuit constant.

When switching the product, matching evaluation with each product and oscillator is necessary.

1.7.3 Unused pins

■ Unused Pins (only for output)

Unconnect the unused output pins.

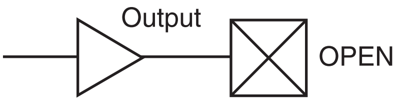


Figure:1.7.2 Unused Pins (only for output)

■ Unused Pins (only for input)

Pull-up (or down) the unused input pins with the resistor, the value of which is typically between 10 kΩ and 100 kΩ.

If the input is unstable, Pch transistor and Nch transistor of input inverter are on, and through current goes to the input circuit. That increases current consumption and becomes noise sources to power supply.

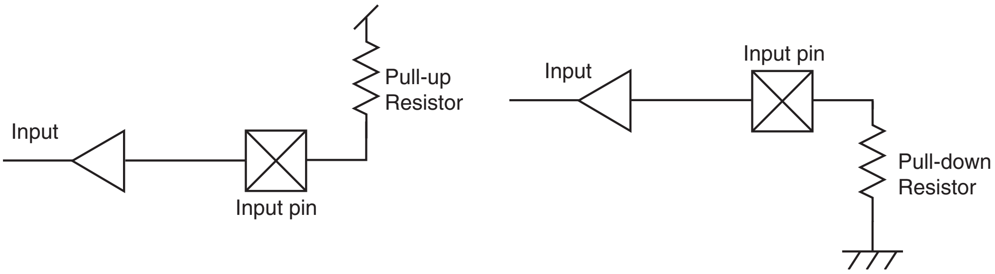


Figure:1.7.3 Unused Pins (only for input)

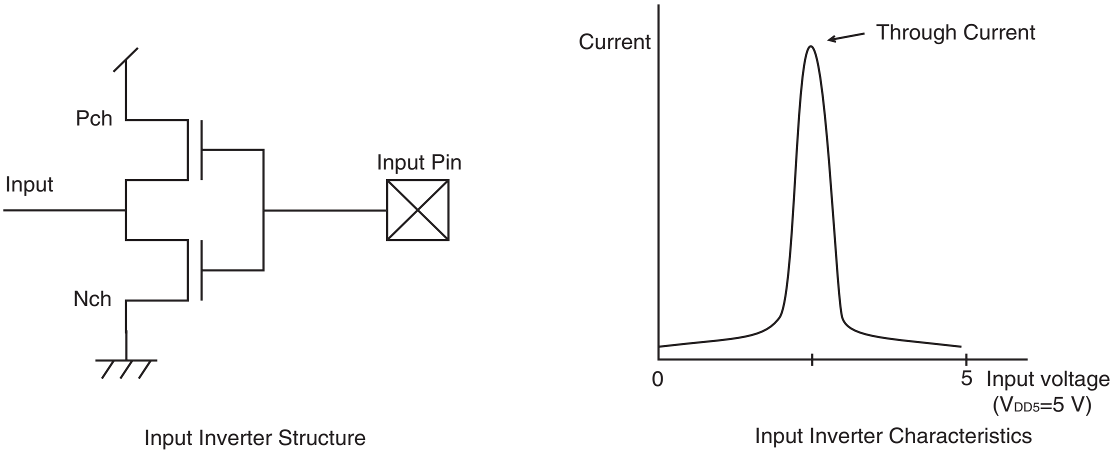


Figure:1.7.4 Structure and Characteristics of Input Inverter

■ Unused Pins (for I/O)

When the direction of unused I/O pin is set to input, pull-up (or down) the pin with the resistor, the value of which is typically between 10 kΩ and 100 kΩ.
When the unused I/O pin is configured as output, it should be left unconnected.

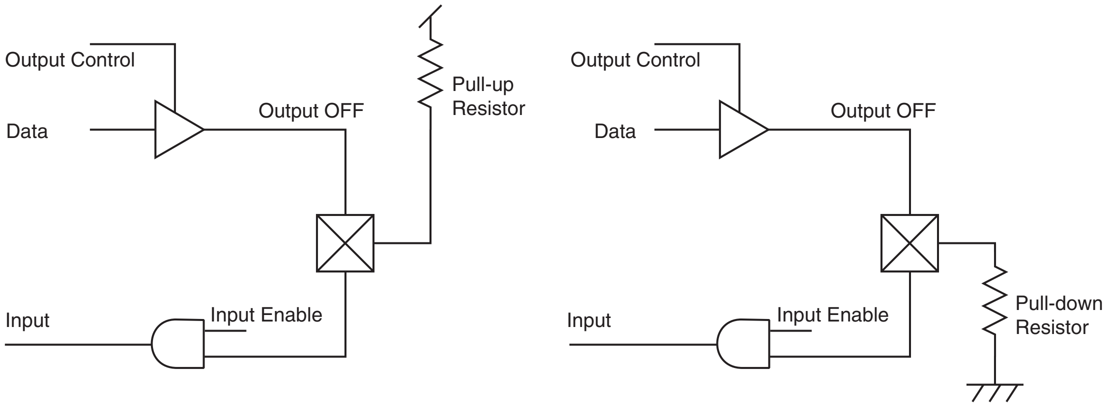


Figure:1.7.5 Unused I/O Pins

Table:1.7.2 Recommended method of each pins

Pin Name	Input/Output	Recommended condition of unused pins	
P00 to P07 P20 to P26 P33 to P35 P43 to P47 P50 to P57 P62 to P67 P70 to P77 P80 to P87 P90 to P94 PA0 to PA7 PB0 to PB3	Input/Output	Input	Pull-up (or down) the pins with the resistor, the value of which is typically between 10 kΩ and 100 kΩ. *1, *2
		Output	Unconnect the pins.
P27	Reset	When a capacitor is to be inserted between P27 and V_{SS}, it is recommended that a discharge diode be placed between P27 and V_{DD5}. *3	
VREF+	Reference power supply	Set the condition of $V_{DD5} = V_{REF+}$.	
ATRST	Input	Pull-up (when auto reset function is enabled) or pull-down (when auto reset function is disabled) the pin with the resistor, the value of which is typically between 10 kΩ and 100 kΩ.	
MMOD	Input	Pull-up (when BOOT mode is enabled) or pull-down (when BOOT mode is disabled) the pin with the resistor, the value of which is typically between 100 Ω and 1000 Ω.	
DMOD	Input	Pull-up the pin with the resistor, the value of which is typically between 10 kΩ and 100 kΩ.	

*1 When unused pins are not connected, the microcomputer does not have the problem. However, it is easily influenced by the serge or the noise.
Evaluate enough for determining the appropriate configuration.

*2 When pins are unused, set them to the normal port function.

*3 The condition of unused pins should be determined by evaluating enough in consideration of the exogenous noise.

1.7.4 Power Supply

■ The Relation between Power Supply and Input Pin Voltage

Input pin voltage should be supplied only after power supply is on. If this order is reversed the destruction of microcontroller by a large current flow could be occurred.

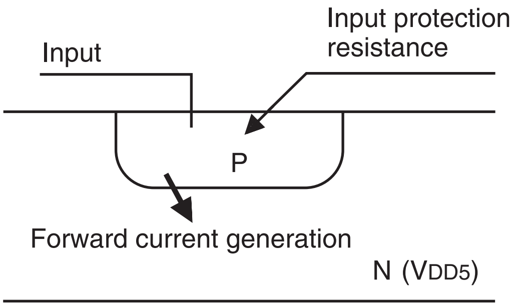


Figure:1.7.6 V_{DD5} and Input Pin Voltage

■ Relation between Power Supply and Reset Input Voltage (When Auto-reset Function is not used)

- Reset
  "Low" level input for 100 µs or longer is required.

To recognize the external reset, time until external voltage reaches to 4.0 V, time for 30 µs after regulator starts operation until it outputs voltage and time for 100 µs or longer are required to input "Low" level to reset pin.

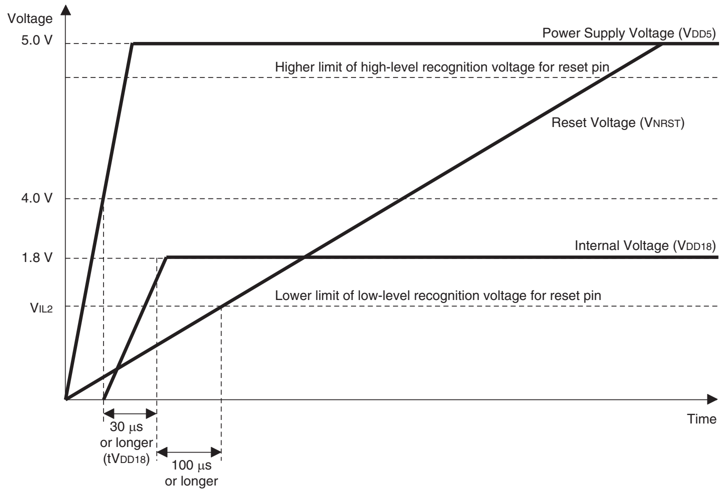
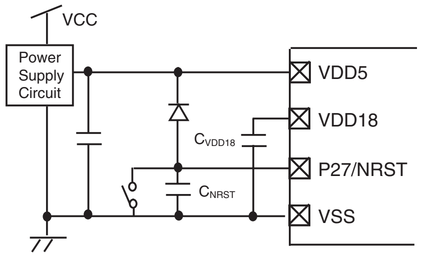


Figure:1.7.7 Power Supply and Reset Input Voltage (No Used Auto Reset)



When,
Power supply (V_{DD5}) = 5.0 V
Abillity to supply power = 50 mA
C_{VDD18} = 1 µF + 0.1 µF
Input current of reset pin = 5.0 V / 50 kΩ = 100 µA
C_{NRST} connects more than the following values.

$tV_{DD18} \approx (1.65\ \text{V} \times\ 1.1\ \mu\text{F})\ /\ 50\ \text{mA} = 36.5\ \mu\text{s}$
$C_{NRST} \approx (36.5\ \mu\text{s} \times 2 \times 100\ \mu\text{A})\ /\ 5.0\ \text{V} \times 0.2 = 7.3\ \text{nF}$

Figure:1.7.8 Power Circuit Example

The circuit connected with the reset pin should be evaluated enough noting the reset generation caused by the exogenous noise for determining the appropriate configuration.

■ Relation between Power Supply and Reset Input Voltage (When Auto-reset function is used)

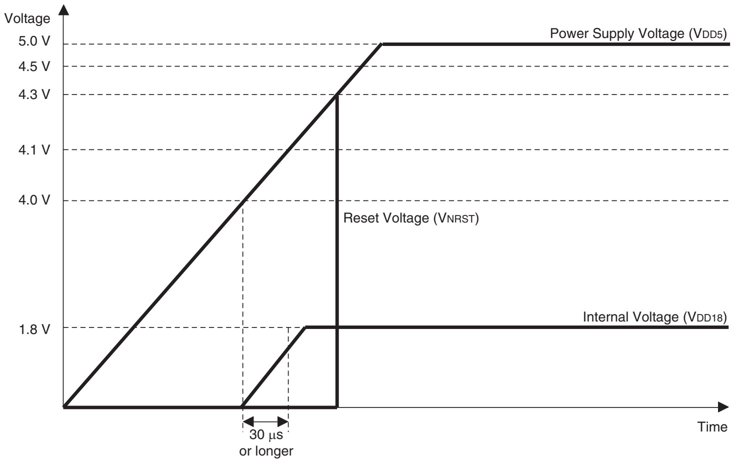


Figure:1.7.9 Power Supply and Reset Input Voltage (Used Auto Reset function)

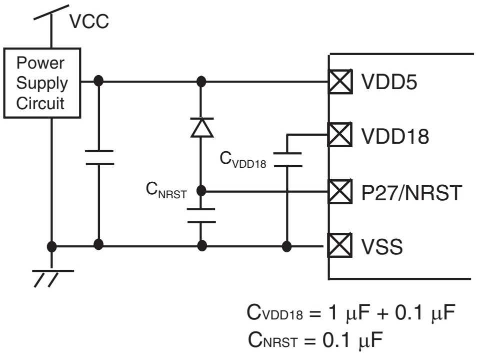


Figure:1.7.10 Power Circuit Example

The circuit connected with the reset pin should be evaluated enough noting the reset generation caused by the exogenous noise for determining the appropriate configuration.

1.7.5 Power Supply Circuit

■ Cautions for Power Circuit Design

The MOS logic such a microcomputer is high speed and high density. So, the power circuit should be designed, taking into consideration of AC line noise, ripple caused by LED driver.

Figure:1.7.11 shows an example for a circuit with V_{DD} (Emitter follower type).

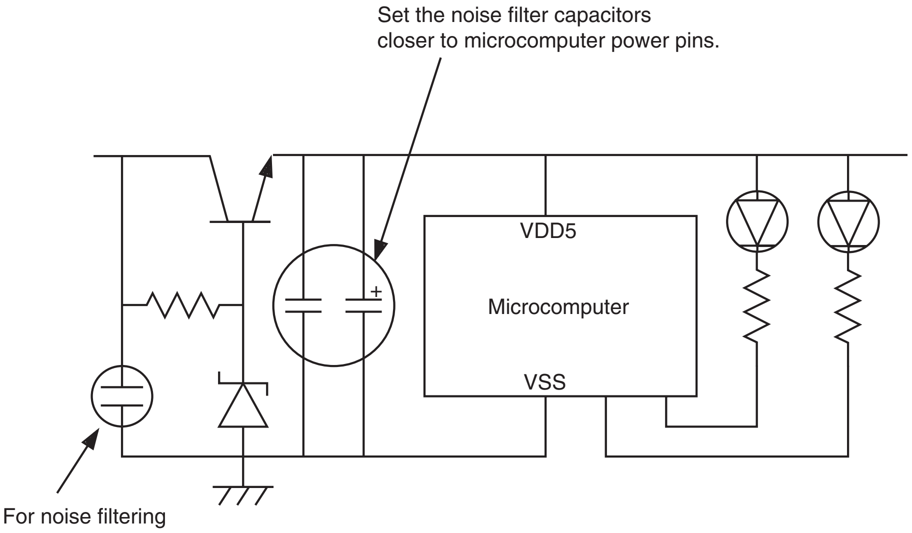

Set the noise filter capacitors
closer to microcomputer power pins.
VDD5
Microcomputer
VSS
+
For noise filtering

Figure:1.7.11 Power Circuit Example (Emitter follower type)

Chapter 2　CPU Basics

2.1 Overview

The MN101E has a flexible and optimized hardware configuration. It is CPU which realizes coexistence of economical efficiency and high-speed operation with a simple and efficient instruction set. Specific features are as follows:

1. Minimized code sizes with instruction lengths based on 4-bit increments:

   The series keeps code sizes down by adopting a basic instruction length of one byte and variable instruction lengths based on 4-bit increments.

2. Minimum execution instruction time is one system clock cycle. (50 ns)

3. Minimized register set that simplifies the architecture and supports C language:

   The instruction set has been determined, depending on the size and capacity of hardware, after on analysis of embedded application programing code and creation code by C language compiler. Therefore, the set is simple instruction using the minimal register set required for C language compiler.

Table:2.1.1 Basic Specifications

Structure	Structure	
	Six registers	Data: 8-bit × 4 Address: 16-bit × 2
	Others	PC: 21-bit PSW: 8-bit SP: 16-bit
Instructions	Number of instructions	39
	Addressing modes	9
	Instruction length	Basic portion: 1 byte (min.) Extended portion: 0.5-byte × n (0≤n≤9)
Basic performance	Internal operating frequency (max)	20 MHz
	Instruction execution	Minimum 1 cycle
	Inter-register operation	Minimum 2 cycle
	Load / store	Minimum 2 cycle
	Conditional branch	2 to 3 cycles
Pipeline	3-stage (instruction fetch, decode, execution)	
Address space	1 MB (Data area: 64 KB (MAX))	
	Instruction/data space	
External bus	Address	20-bit (Maximum)
	Data	8-bit
	Minimum bus cycle	1 system clock cycle
Interrupt	Vector interrupt	3 interrupt levels
Low-power consumption mode	STOP mode	
	HALT mode	

2.1.1 Block Diagram

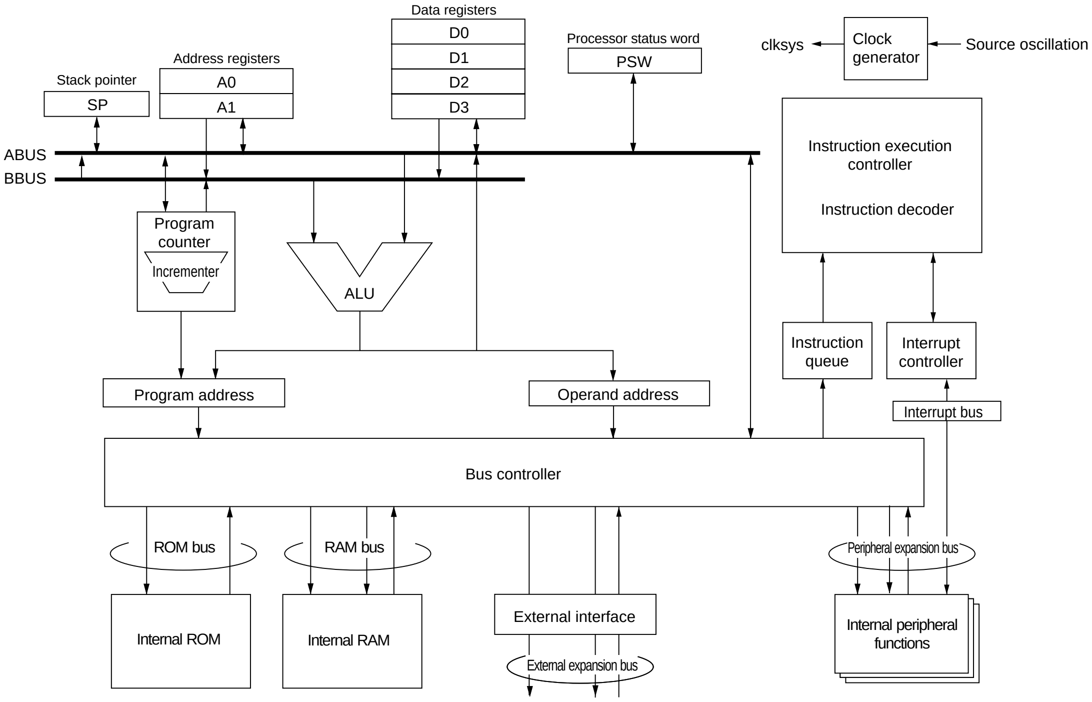


Figure:2.1.1 CPU Block Diagram

Table:2.1.2 Block Diagram and Function

Clock generator	Uses a clock oscillator circuit driven by an external crystal or ceramic resonator to supply clock signals to CPU blocks.
Program counter	Generates addresses for the instructions to be inserted into the instruction queue. Normally incremented by sequencer indication, but may be set to branch destination address or ALU operation result when branch instructions or interrupts occur.
Instruction queue	Stores up to 2 bytes of pre-fetched instructions.
Instruction decoder	Decodes the instruction queue, sequentially generates the control signals needed for instruction execution, and executes the instruction by controlling the blocks within the chip.
Instruction execution controller	Controls CPU block operations in response to the result decoded by the instruction decoder and interrupt requests.
ALU	Executes arithmetic operations, logic operations, shift operations, and calculates operand addresses for register relative indirect addressing mode.
Internal ROM, RAM	Assigned to the execution program, data and stack region.
Address register	Stores the addresses specifying memory for data transfer. Stores the base address for register relative indirect addressing mode.
Data register	Holds data for operations. Two 8-bit registers can be connected to form a 16-bit register.
Interrupt controller	Detects interrupt requests from peripheral functions and requests CPU shift to interrupt processing.
Bus controller	Controls connection of CPU internal bus and CPU external bus. Includes bus usage arbitration function.
Internal peripheral functions	Includes peripheral functions (timer, serial interface, A/D converter, D/A converter, etc.). Peripheral functions vary depending on the model.

2.1.2 CPU Control Registers

This LSI locates the peripheral circuit registers in memory space (0x03DF0 to 0x03FFF) with memory mapped I/O. CPU control registers are also located in this memory space.

Table:2.1.3 CPU Control Registers

Table remarks √: With function -: Without function

Registers	Address	R/W	Function	Page	MN101EF A8/A3	MN101EF A7/A2
CPUM	0x03F00	R/W	CPU mode control register	III-16	√	√
MEMCTR	0x03F01	R/W	Memory control register	II-35	√	√
Reserved	0x03F04	-	(For test)	-	√	√
Reserved	0x03F0F	-	(For test)	-	√	√
Reserved	0x03FE0	-	(For debugger)	-	√	√
NMICR	0x03FE1	R/W	Non-maskable interrupt control register	IV-20	√	√
xxxICR	0x03FE2 to 0x03FFD	R/W	Maskable interrupt control register	IV-21 to IV-30	√	√
Reserved	0x03FFF	-	(For reading interrupt vector data on interrupt process)	-	√	√

Do not access the reserved address (read / write).

2.1.3 Instruction Execution Controller

The instruction execution controller consists of four blocks: memory, instruction queue, instruction registers, and instruction decoder.

Instructions are fetched in 1-byte units, and temporarily stored in the 2-byte instruction queue. Transfer is made in 1-byte or half-byte units from the instruction queue to the instruction register to be decoded by the instruction decoder.

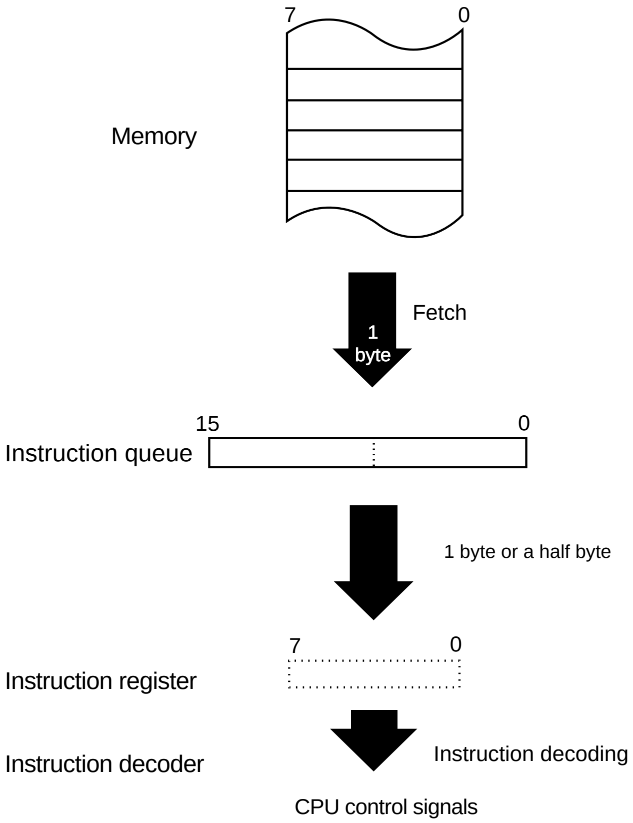


Figure:2.1.2 Instruction Execution Controller Configuration

2.1.4 Pipeline Process

Pipeline process means that reading and decoding are executed at the same time on different instructions, then instructions are executed without stopping. Pipeline process enables instruction execution continuously and faster. This process is executed with instruction queue and instruction decoder.

Instruction queue is buffer that fetches the second instruction in advance. That is controlled to fetch the next instruction when instruction queue is empty at each cycle on execution. At the last cycle of instruction execution, the first word (operation code) of executed next instruction is stored to instruction register. At that time, the next operand or operation code is fetched to instruction queue, so that the next instruction can be executed immediately, even if register direct (da) or immediate data (imm) is needed at the first cycle of the next instruction execution. But on some other instruction such as branch instruction, instruction queue becomes empty on the time that the next operation code to be executed is stored to instruction register at the last cycle. Therefore, only when instruction queue is empty, and direct address (da) or immediate data (imm) are needed, instruction queue keeps waiting for a cycle.

Instruction queue is controlled automatically by hardware so that there is no need to be controlled by software. But when instruction execution time is estimated, operation of instruction queue should be take into consideration. Instruction decoder generates control signal at each cycle of instruction execution by micro program control. Instruction decoder uses pipeline process to decode instruction queue at one cycle before control signal is needed.

2.1.5 Registers for Address

Registers for address include program counter (PC), address registers (A0, A1), and stack pointer (SP)

■ Program Counter (PC)

This register gives the address of the currently executing instruction. It is 1 MB bits wide to provide access to a 21 address space in half byte(4-bit increments). The LSB of the program counter is used to indicate half byte instruction. The program counter after reset is stored from the value of vector table at the address of 0x04000.

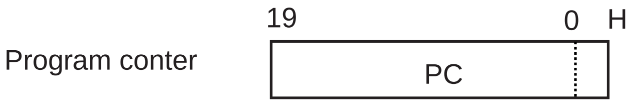


Figure:2.1.3 Program Counter

■ Address Registers (A0, A1)

These registers are used as address pointers specifying data locations in memory. They support the operations involved in address calculations (i.e. addition, subtraction and comparison). Those pointers are 2 bytes data. Transfers between these registers and memory are always in 16-bit units. Either odd or even address can be transferred. At reset, the value of address register is undefined.

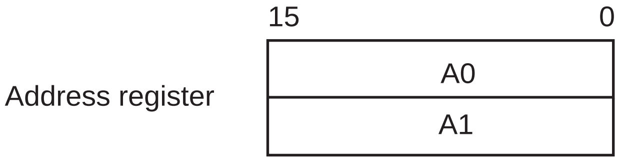


Figure:2.1.4 Address Registers

■ Stack Pointer (SP)

This register gives the address of the byte at the top of the stack. It is decremented during push operations and incremented during pop operations. At reset, the value of SP is undefined.

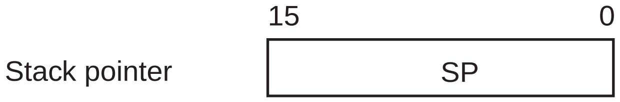


Figure:2.1.5 Stack Pointer

2.1.6 Registers for Data

Registers for data include four data registers (D0, D1, D2, D3).

- Data Registers (D0, D1, D2, D3)

Data registers D0 to D3 are 8-bit general-purpose registers that support all arithmetic, logical and shift operations. All registers can be used for data transfers with memory.The four data registers may be paired to form the 16-bit data registers DW0 (D0+D1) and DW1 (D2+D3). At reset, the value of Dn is undefined.

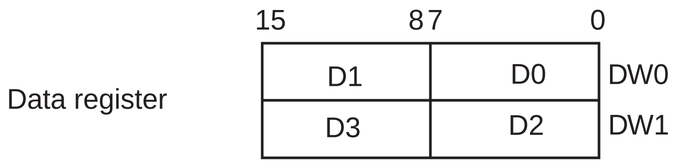


Figure:2.1.6 Data Registers

2.1.7 Processor Status Word

Processor status word (PSW) is an 8-bit register that stores flags for operation results, interrupt mask level, and maskable interrupt enable flag. PSW is automatically pushed onto the stack when an interrupt occurs and is automatically popped when return from the interrupt service routine.

■ Processor Status Word (PSW)

bp	7	6	5	4	3	2	1	0
Flag	BKD	MIE	IM1	IM0	VF	NF	CF	ZF
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7	BKD	Bank disable flag 0: Bank addressing is enabled. 1: Bank addressing is disabled.
6	MIE	Maskable interrupt enable 0: All maskable interrupts are disabled. 1: (xxxLVn,xxxIE) for each interrupt is enabled.
5 to 4	IM1 IM0	Interrupt mask level Controls maskable interrupt acceptance. [Chapter 2 Table:2.1.4 Interrupt Mask Level and Interrupt Acceptance]
3	VF	Overflow flag 0: No overflow occurred. 1: Overflow occured.
2	NF	Negative flag 0: MSB of operation results is "0". 1: MSB of operation results is "1".
1	CF	Carry flag 0: No carry or borrow from MSB occurred. 1: A carry or a borrow from MSB occures.
0	ZF	Zero flag 0: Operation result is not "0". 1: Operation result is "0".

■ Zero Flag (ZF)

Zero flag (ZF) is set to "1", when all bits are '0' in the operation result. Otherwise, zero flag is cleared to "0".

■ Carry Flag (CF)

Carry flag (CF) is set to "1", when a carry from or a borrow to the MSB occurs. Carry flag is cleared to "0", when no carry or borrow occurs.

■ Negative Flag (NF)

Negative flag (NF) is set to "1" when MSB is '1' and reset to "0" when MSB is '0'. Negative flag is used to handle a signed value.

■ Overflow Flag (VF)

Overflow flag (VF) is set to "1", when the operation results overflow as a signed value. Otherwise, overflow flag is cleared to "0". Overflow flag is used to handle a signed value.

■ Interrupt Mask Level (IM1 and IM0)

Interrupt mask level (IM1 and IM0) controls the maskable interrupt acceptance in accordance with the interrupt factor interrupt priority for the interrupt control circuit in the CPU. The two-bit control flag defines levels '0' to '3'. Level 0 is the highest mask level. The interrupt request will be accepted only when the level set in the interrupt level flag (xxxLV1 and xxxLV0) of the interrupt control register (xxxICR) is higher than the interrupt mask level. When the interrupt is accepted, the value of accepted interrupt level flag is set to IM1-IM0, and interrupts whose mask levels are the same or lower are rejected during the accepted interrupt processing.

Table:2.1.4 Interrupt Mask Level and Interrupt Acceptance

	Interrupt mask level		Priority	Acceptable interrupt level
	IM1	IM0		
Mask level 0	0	0	High	Non-maskable interrupt (NMI) only
Mask level 1	0	1	-	NMI, level 0
Mask level 2	1	0	-	NMI, level 0 to 1
Mask level 3	1	1	Low	NMI, level 0 to 2

■ Maskable Interrupt Enable (MIE)

Maskable interrupt enable flag (MIE) enables/disables acceptance of maskable interrupts by the CUP's internal interrupt acceptance circuit. A '1' enables maskable interrupts; a '0' disables all maskable interrupts regardless of the interrupt mask level (IM1-IM0) setting in PSW. This flag is not changed by interrupts.

■ Bank disable flag (BKD)

Bank disable flag (BKD) enables/disables bank addressing of 64 KB unit. When this flag is set to "0", bank addressing is enabled and you can access to total 16 banks by setting the bank register value. When this flag is set to "1", bank addressing is disabled and the only area you can access is the first 64 KB. On an interrupt generation, BKD flag is automatically set to "1" and bank addressing is disabled. At returning from interrupt service routine, the value of BKD flag is returned to the previous one. (before the interrupt generation)

To enable bank addressing in an interrupt service routine, reset the BKD flag to "0" before accessing to data.

Write to the interrupt control register (xxxICR) after setting the maskable interrupt enable flag (MIE) of processor status word (PSW) to disable all maskable interrupts. (MIE flag is set to "0".) There's no guarantee of proper operation when writing is executed to the interrupt control register with setting MIE flag to "1".

2.1.8 Address Space

The address space of this LSI is 1 MB (max). The instruction and data areas are in the same area.

The instruction area can be used as linear address space. The data area needs bank specification in every 64 KB. (The initial value is first 64 KB space). The data described in this section includes RAM data and ROM table data.

The data area consists of an area of 256 bytes that supports efficient accesses with RAM short addressing and an area of 256 bytes that supports efficient accesses with I/O short addressing.

The memory control register controls the operation of the memory to be expanded.

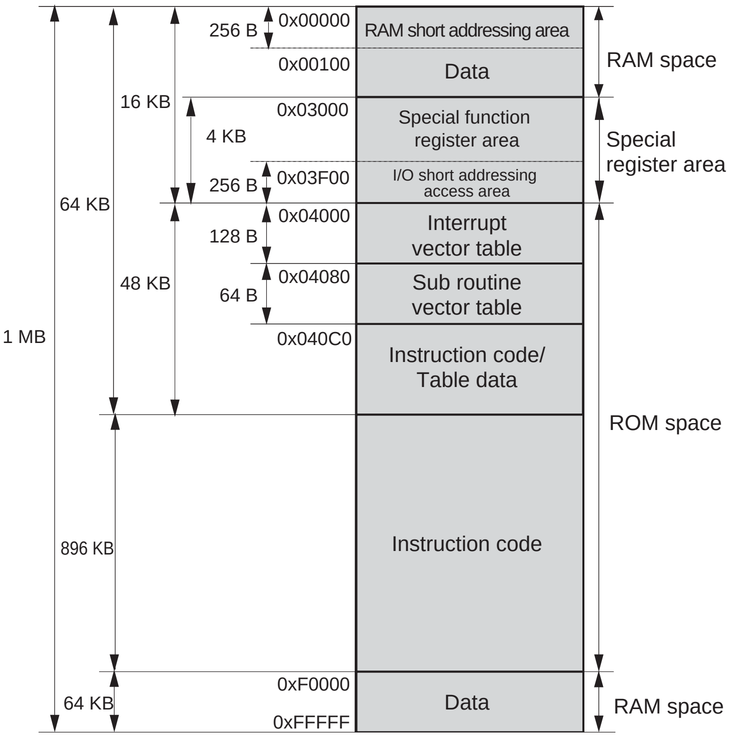


Figure:2.1.7 Address Space

2.1.9 Addressing Modes

This LSI supports the nine addressing modes. Each instruction uses a combination of the following addressing

1. Register direct
2. Immediate
3. Register indirect
4. Register relative indirect
5. Stack relative indirect
6. Absolute
7. RAM short
8. I/O short
9. Handy

These addressing modes are well-suited for C language compilers. All of the addressing modes can be used for data transfer instructions. In modes that allow half-byte addressing, the relative value can be specified in half-byte (4-bit) increments, so that instruction length can be shorter. Handy addressing reuses the last memory address accessed. There are three instructions that can use this mode: MOV Dn, (HA), MOVW DWn, (HA), MOVW An, (HA). Combining handy addressing with absolute addressing reduces code size. For transfer data between memory, 8 addressing modes; register indirect, register relative indirect, stack relative indirect, absolute, RAM short, I/O short, handy can be used. For operation instruction, register direct and immediate can be used. Refer to instruction's manual for the MN101E series.

This LSI is designed for 8-bit data access. When 16-bit data access is carried out, 8-bit data access is performed twice from the lower address.
It is possible to transfer data in 16-bit increments with odd or all even addresses.

Table:2.1.5 Address Space

Addressing mode		Effective address	Explanation
Register direct	Dn/DWn An/SP PSW	-	Directly specifies the register. Only internal registers can be specified.
Immediate	imm4/imm8 imm16	-	Directly specifies the operand or mask value appended to the instruction code.
Register indirect	(An)	15 0 An	Specifies the address using an address register.
Register relative indirect	(d8,An)	15 0 An+d8	Specifies the address using an address register with 8-bit displacement.
	(d16,An)	15 0 An+d16	Specifies the address using an address register with 16-bit displacement.
	(d4,PC) (branch instructions only)	17 0 H PC+d4 *1	Specifies the address using the program counter with 4-bit displacement and H bit.
	(d7,PC) (branch instructions only)	17 0 H PC+d7 *1	Specifies the address using the program counter with 7-bit displacement and H bit.
	(d11,PC) (branch instructions only)	17 0 H PC+d11 *1	Specifies the address using the program counter with 11-bit displacement and H bit.
	(d12,PC) (branch instructions only)	17 0 H PC+d12 *1	Specifies the address using the program counter with 12-bit displacement and H bit.
	(d16,PC) (branch instructions only)	17 0 H PC+d16 *1	Specifies the address using the program counter with 16-bit displacement and H bit.
Stack relative indirect	(d4,SP)	15 0 SP+d4	Specifies the address using the stack pointer with 4-bit displacement.
	(d8,SP)	15 0 SP+d8	Specifies the address using the stack pointer with 8-bit displacement.
	(d16,SP)	15 0 SP+d16	Specifies the address using the stack pointer with 16-bit displacement.
Absolute	(abs8)	7 0 abs8	Specifies the address using the operand value appended to the instruction code.Optimum operand length can be used to specify the address.
	(abs12)	11 0 abs12	
	(abs16)	15 0 abs16	
	(abs18) (branch instructions only)	17 0 H abs18 *1	
	(abs20) (branch instructions only)	19 0 H abs20 *1	
RAM short	(abs8)	7 0 abs8	Specifies an 8-bit offset from the address x'00000'.
I/O short	(io8)	15 0 IOTOP+io8	Specifies an 8-bit offset from the top address (x'03F00') of the special function register area.
Handy	(HA)	-	Reuses the last memory address accessed and is only available with the MOV and MOVW instructions. Combined use with absolute addressing reduces code size.

*1 H: half-byte bit

2.1.10 Machine Clock

Machine clock is generated based on the system clock (fs) dividing the source oscillation frequency. The machine clock is the base timing for control of CPU.

- Internal Memory Access

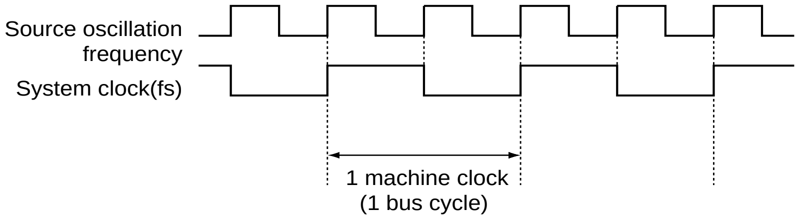


Figure:2.1.8 Machine Clock of Internal Memory Access

Division ratio of system clock (fs) differs depending on the CPUM register settings.
Refer to [Chapter 3 Clock Control].

2.2 Memory Space

2.2.1 Memory Mode

ROM is the read only area and RAM is the memory area which is readable/writable of data. In addition to these, peripheral resources such as memory-mapped special registers are allocated.

This LSI supports only the single chip mode, in which CPU accesses the internal memory.

Always set bp4 of the MEMCTR register to "0". When setting "1", there's no guarantee of proper operation.

Table:2.2.1 Memory Mode Setup

Memory mode	MMOD
Single chip mode	Low

Do not change the settings of MMOD pin even after reset is released.

For MMOD setup to rewrite the flash memory, refer to [Chapter 16 16.5 User Mode Microcontroller Rewriting], [Chapter 16 16.6 BOOT Mode Microcontroller Rewriting], [Chapter 16 16.7 Appendix].

2.2.2 RAM Space

■ RAM Space

MN101E series has maximum 64 KB of RAM space.
RAM space is divided to be allocated to the address space. Mirror RAM space is provided for effective utilization of the divided RAM spaces.
RAM space: 0x00000 to 0x02FFF (12 KB) + 0xF3000 to 0xFFFFF (52 KB) (maximum 64 KB)

Mirror RAM space: 0xF0000 to 0xF3DFF = 0x00000 to 0x03DFF (Mapped to same RAM space)

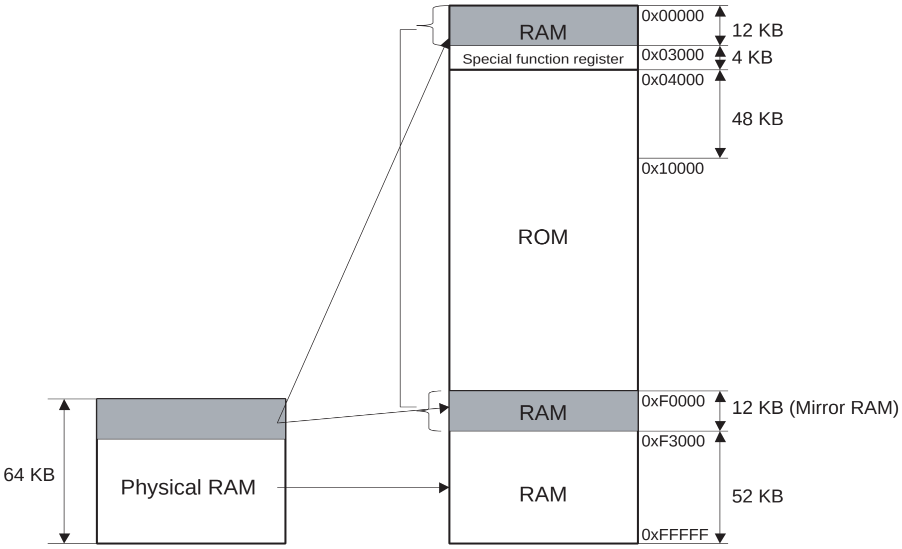


Figure:2.2.1 RAM Space

■ How to use mirror RAM Space

Sub routine A

Address bank 15

mov x'0F', (SBNKR): Source side

mov x'0F', (DBNKR): Destination side

Transfer data 15 between memories (1)

mov (x'XYZZ'), dn

mov dn, (x'ABCD'): x'XYZZ' → x'ABCD'

(x'ABCD', x'XYZZ' are address of abs16.)

Sub routine B (Address bank 0)

mov (x'ABCD'), d1: Use mirror function (2)

Execute the same access ignoring the upper 4 bits.

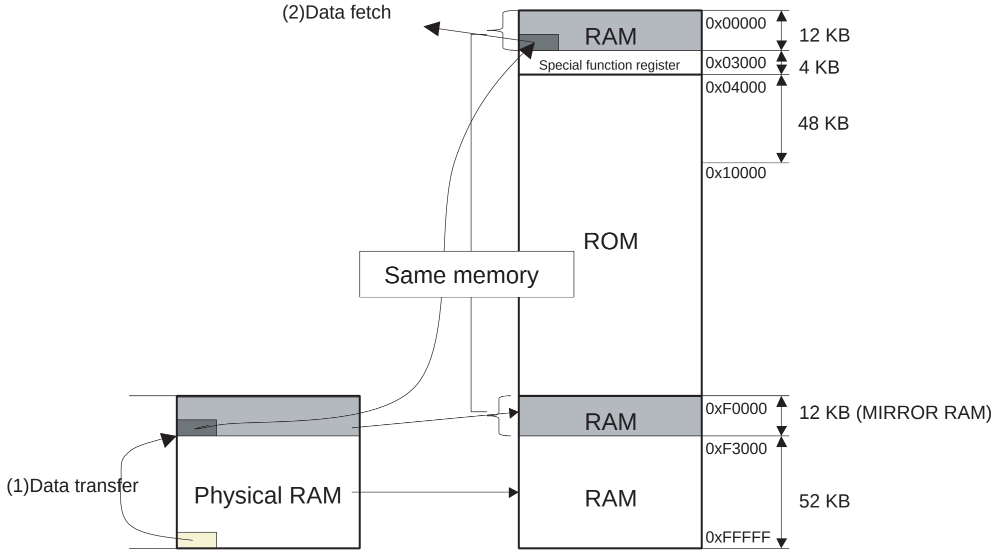


Figure:2.2.2 How to use mirror RAM Space (Example)

2.2.3 Single-chip Mode (MN101EFA8G/A7G/A3G/A2G)

In single-chip mode, the system consists of only internal memory. This is the optimized memory mode and allows construction of systems with the highest performance. The single-chip mode uses only internal ROM and internal RAM. The MN101E series devices offer up to 64 KB of RAM and up to 944 KB of ROM. This LSI offers 6 KB of RAM and 128 KB of ROM.

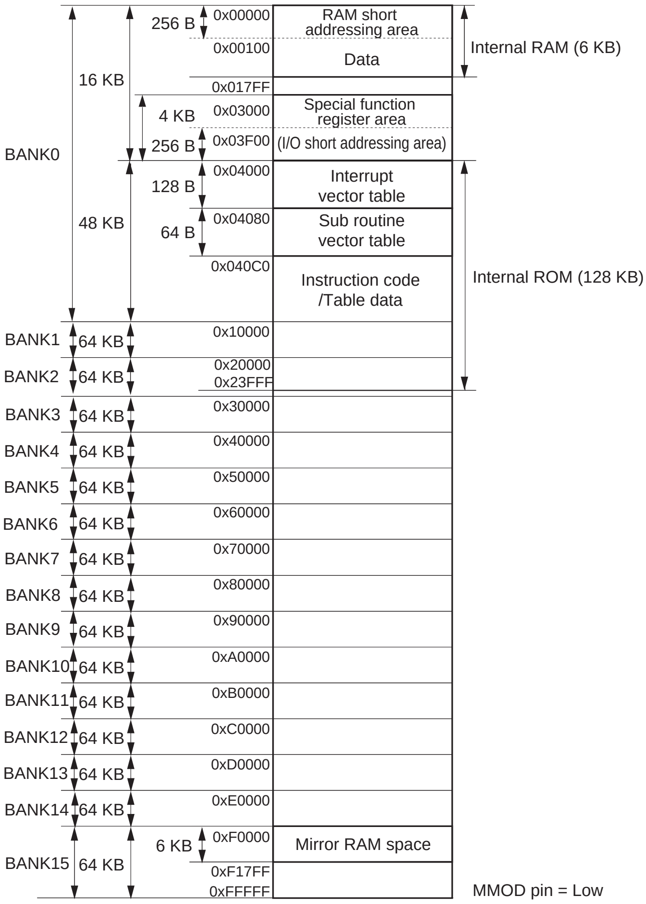


Figure:2.2.3 Single-chip Mode (MN101EFA8G/A7G/A3G/A2G)

The value of internal RAM is uncertain at power-on. It needs to be initialized before using.

There's no guarantee of proper operation when an access is executed to the non-implemented space where a memory (ROM/RAM), a special function register, or others are not arranged.

2.2.4 Single-chip Mode (MN101EFA8D/A7D/A3D/A2D)

In single-chip mode, the system consists of only internal memory. This is the optimized memory mode and allows construction of systems with the highest performance. The single-chip mode uses only internal ROM and internal RAM. The MN101E series devices offer up to 64 KB of RAM and up to 944 KB of ROM. This LSI offers 4 KB of RAM and 64 KB of ROM.

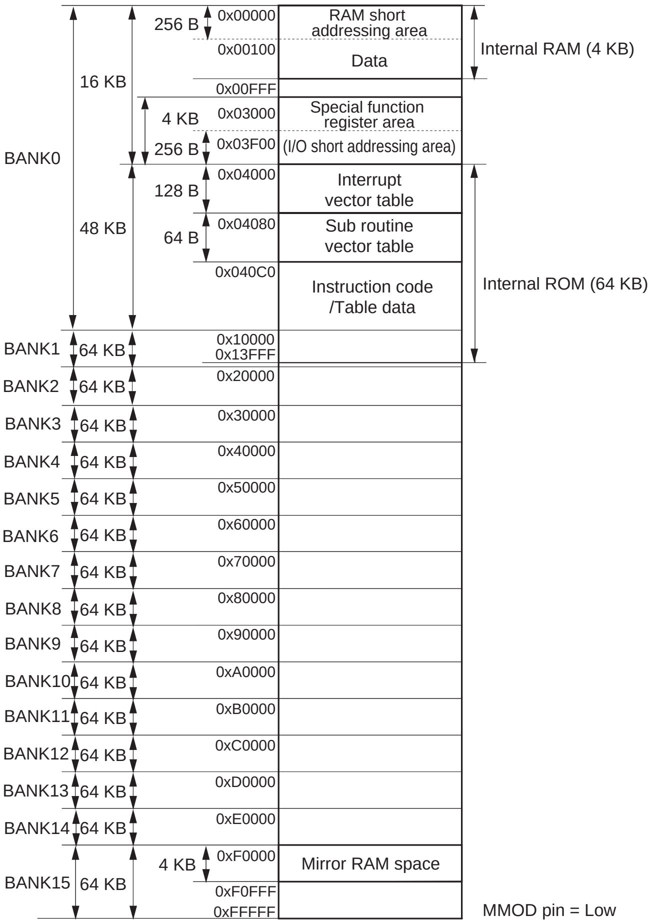


Figure:2.2.4 Single-chip Mode (MN101EFA8D/A7D/A3D/A2D)

The value of internal RAM is uncertain at power-on. It needs to be initialized before using.

There's no guarantee of proper operation when an access is executed to the non-implemented space where a memory (ROM/RAM), a special function register, or others are not arranged.

2.2.5 Bank Function

CPU of the MN101E series basically has 64 KB data area. On this LSI, data area can be expanded up to 16 banks (1 MB) with a unit of 64 KB for bank function.

Bank function can be used by setting the proper bank area to the bank register for source address (SBNKR) or the bank register for destination address (DBNKR). At reset, both SBNKR register and DBNKR register indicate bank 0. Bank function is valid after setting bank disable flag (BKD), at bit-7 of processor status word (PSW) to “0”.

When SBNKR and DBNKR registers are operated in interrupt processing, pushing into the stack or popping must be done by program, if needed.

Table:2.2.2 Address Range

SBA3 (DBA3)	SBA2 (DBA2)	SBA1 (DBA1)	SBA0 (DBA0)	Bank area	Address Range
0	0	0	0	bank 0	0x00000 to 0x0FFFF
0	0	0	1	bank 1	0x10000 to 0x1FFFF
0	0	1	0	bank 2	0x20000 to 0x2FFFF
0	0	1	1	bank 3	0x30000 to 0x3FFFF
0	1	0	0	bank 4	0x40000 to 0x4FFFF
0	1	0	1	bank 5	0x50000 to 0x5FFFF
0	1	1	0	bank 6	0x60000 to 0x6FFFF
0	1	1	1	bank 7	0x70000 to 0x7FFFF
1	0	0	0	bank 8	0x80000 to 0x8FFFF
1	0	0	1	bank 9	0x90000 to 0x9FFFF
1	0	1	0	bank 10	0xA0000 to 0xAFFFF
1	0	1	1	bank 11	0xB0000 to 0xBFFFF
1	1	0	0	bank 12	0xC0000 to 0xCFFFF
1	1	0	1	bank 13	0xD0000 to 0xDFFFF
1	1	1	0	bank 14	0xE0000 to 0xEFFFF
1	1	1	1	bank 15	0xF0000 to 0xFFFFF

When changing a bank during interrupt processing, conduct saving or returning processing by software according to need.

While bank function is valid, I/O short instruction should be used when accessing 0x03F00 to 0x03FFF in the special function register area (0x03000 to 0x03FFF). When accessing the memory space 0x13F00 to 0x13FFF, 0x23F00 to 0x23FFF, 0x33F00 to 0x33FFF, 0x43F00 to 0x43FFF, 0x53F00 to 0x53FFF, 0x63F00 to 0x63FFF, 0x73F00 to 0x73FFF, 0x83F00 to 0x83FFF, 0x93F00 to 0x93FFF, 0xA3F00 to 0xA3FFF, 0xB3F00 to 0xB3FFF, 0xC3F00 to 0xC3FFF, 0xD3F00 to 0xD3FFF, 0xE3F00 to 0xE3FFF, 0xF3F00 to 0xF3FFF, both instructions of register indirect and register relative indirect should be used.
[Chapter 2 2.1.9 Addressing Modes]

Set the stack area to bank 0. The provided C-compiler for this series does not support bank function.

Our linker supports the function that prevents data from straddling over bank boundaries. See "MN101E Series Cross-assembler User's Manual" for details.

■ Bank Register for Source Address (SBNKR: 0x03F0A)

The SBNKR register is used to specify bank areas for loading instructions from memory to register. Once this register is specified, bank control is valid for all addressing modes except I/O short instruction and stack relative indirect instruction.

[Chapter 2 2.1.9 Addressing Modes]

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	SBA3	SBA2	SBA1	SBA0
At reset	-	-	-	-	0	0	0	0
Access	-	-	-	-	R/W	R/W	R/W	R/W

bp	Flag	Description
7 to 4	-	-
3 to 0	SBA3 SBA2 SBA1 SBA0	Bank for source address selection 0000: bank 0 0001: bank 1 0010: bank 2 0011: bank 3 0100: bank 4 0101: bank 5 0110: bank 6 0111: bank 7 1000: bank 8 1001: bank 9 1010: bank 10 1011: bank 11 1100: bank 12 1101: bank 13 1110: bank 14 1111: bank 15

■ Bank Register for Destination Address (DBNKR: 0x03F0B)

The DBNKR register is used to specify bank area for storing instruction from register to memory. Once this register is specified, bank control is valid for all addressing modes except I/O short instruction, stack relative indirect instruction and bit manipulation instruction.

[Chapter 2 2.1.9 Addressing Modes]

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	DBA3	DBA2	DBA1	DBA0
At reset	-	-	-	-	0	0	0	0
Access	-	-	-	-	R/W	R/W	R/W	R/W

bp	Flag	Description
7 to 4	-	-
3 to 0	DBA3 DBA2 DBA1 DBA0	Bank selection for destination address 0000: bank 0 0001: bank 1 0010: bank 2 0011: bank 3 0100: bank 4 0101: bank 5 0110: bank 6 0111: bank 7 1000: bank 8 1001: bank 9 1010: bank 10 1011: bank 11 1100: bank 12 1101: bank 13 1110: bank 14 1111: bank 15

Read-modify-write instruction such as bit manipulation (BSET, BCLR, BTST) depends on the value of SBNKR register for both reading and writing.

2.2.6 Special Function Registers

This LSI locates the special function registers (I/O spaces) in memory space with addresses between 0x03E00 to 0x03FFF. The locations of the special function registers for this LSI are shown below.

Do not access (read/write) to the "Reserved" address. In that case, proper operation is not guaranteed.

Each register has a different setting according to bp. Refer to the detailed description for its usage.

Table:2.2.3 Register Map (MN101EFA8)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
03DAx	TS0TMD	TS0CKMD	TS0TCHSEL	TS0RESULT	TS0ERROR	TS0ADCNT										
03DBx	TS0CH0 EXDATL	TS0CH0 EXDATH	TS0CH1 EXDATL	TS0CH1 EXDATH	TS0CH2 EXDATL	TS0CH2 EXDATH	TS0CH3 EXDATL	TS0CH3 EXDATH	TS0CH4 EXDATL	TS0CH4 EXDATH	TS0CH5 EXDATL	TS0CH5 EXDATH	TS0CH6 EXDATL	TS0CH6 EXDATH	TS0CH7 EXDATL	TS0CH7 EXDATH
03DCx	TS0CH0 CTDATL	TS0CH0 CTDATH	TS0CH1 CTDATL	TS0CH1 CTDATH	TS0CH2 CTDATL	TS0CH2 CTDATH	TS0CH3 CTDATL	TS0CH3 CTDATH	TS0CH4 CTDATL	TS0CH4 CTDATH	TS0CH5 CTDATL	TS0CH5 CTDATH	TS0CH6 CTDATL	TS0CH6 CTDATH	TS0CH7 CTDATL	TS0CH7 CTDATH
03DDx	TS1TMD	TS1CKMD	TS1TCHSEL	TS1RESULT	TS1ERROR	Reserved										
03DEx	TS1CH0 EXDATL	TS1CH0 EXDATH	TS1CH1 EXDATL	TS1CH1 EXDATH	TS1CH2 EXDATL	TS1CH2 EXDATH	TS1CH3 EXDATL	TS1CH3 EXDATH								
03DFx	TS1CH0 CTDATL	TS1CH0 CTDATH	TS1CH1 CTDATL	TS1CH1 CTDATH	TS1CH2 CTDATL	TS1CH2 CTDATH	TS1CH3 CTDATL	TS1CH3 CTDATH								
03E0X	PWM-MDL	PWM-MDH	PWM-SELL	PWM-SELH	PWM-SETL	PWM-SETH	TCMPAL	TCMPAH	TCMPBL	TCMPBH	TCMPCL	TCMPCH	OUTMD	DTMSET	DTMSET1	
03E1X	PWM-BCL	PWM-BCH	BCSTR	PWMOFFL	PWMOFFH	IRQCULL	PWMTM-CNT	RELCTR	PWMODR	REL-STAT	PWMCMP1	PWMCMP2				
03E2x																
03E3x																
03E4x																
03E5X	PRTKEY						OSCLOCK									
03E6X	TS0ATCNT0	TS0ATCNT1	TS0ATTRC1	TS0ATMAP0L	TS0ATMAP0M		TS0ATREGAP		TS1ATCNT0	TS1ATCNT1	TS1ATTRC1	TS1ATMAP0L	TS1ATMAP0M		TS1ATREGAP	
03E7X	P0OUT		P2OUT	P3OUT	P4OUT	P5OUT	P6OUT	P7OUT	P8OUT	P9OUT	PAOUT	PBOUT				
03E8X	P0IN		P2IN	P3IN	P4IN	P5IN	P6IN	P7IN	P8IN	P9IN	PAIN	PBIN				
03E9X	P0DIR		P2DIR	P3DIR	P4DIR	P5DIR	P6DIR	P7DIR	P8DIR	P9DIR	PADIR	PBDIR				
03EAX	P0PLU		P2PLU	P3PLUD	P4PLU	P5PLU	P6PLU	P7PLU	P8PLU	P9PLUD	PAPLU	PBPLUD				SELUD
03EBX	P0OMD1					P5OMD	P6OMD		P8OMD		PAOMD					SELUD2
03ECX	P0OMD2			P3IMD						P9IMD	PAIMD	PBIMD				Reserved
03EDX																
03EEX	LEDCNT															
03EFX	P0ODC			P3ODC	P4ODC	P5ODC	P6ODC	P7ODC							Reserved	Reserved
03F0X	CPUM	MEM-CTR	WDCTR	DLYCTR	Reserved		HAND-SHAKE	AUCTR		Reserved	SBNKR	DBNKR	Reserved	Reserved		Reserved
03F1X	SC0SEL	SC0MD0	SC0MD1	SC0MD2	SC0MD3	SC0STR	RXBUF0	TXBUF0		RST-FACT	OSCCNT	RCCNT	OSC-SCNT		EDGDT	PLLCNT
03F2X	SC1SEL	SC1MD0	SC1MD1	SC1MD2	SC1MD3	SC1STR	RXBUF1	TXBUF1	SCINT-SEL		Reserved	Reserved				
03F3X	SC2SEL	SC2MD0	SC2MD1	SC2MD2	SC2MD3	SC2STR	RXBUF2	TXBUF2						Reserved	KEYT3_1IMD	KEYT3_2IMD
03F4X															IRQEX-PEN	IRQEX-PDT
03F5X	SC4MD0	SC4MD1	SC4MD2	SC4MD3	SC4AD0	SC4AD1	SC4STR0	SC4STR1	RXBUF4	TXBUF4	SC4SEL				STB_EDG	STB_MSK
03F6X	TM0BC	TM1BC	TM0OC	TM1OC	TM0MD	TM1MD	CK0MD	CK1MD	TM2BC	TM3BC	TM2OC	TM3OC	TM2MD	TM3MD	CK2MD	CK3MD
03F7X		TMABC		TMAOC		TMAMD1		TMAMD2	TM6BC	TM6OC	TM6MD	TBCLR	TM6BEN		TM7MD4	TM8MD4
03F8X	TM7BCL	TM7BCH	TM7OC1L	TM7OC1H	TM7PR1L	TM7PR1H	TM7ICL	TM7ICH	TM7MD1	TM7MD2	TM7OC2L	TM7OC2H	TM7PR2L	TM7PR2H		

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
03F9X	TM8BCL	TM8BCH	TM8OC1L	TM8OC1H	TM8PR1L	TM8PR1H	TM8ICL	TM8ICH	TM8MD1	TM8MD2	TM8OC2L	TM8OC2H	TM8PR2L	TM8PR2H	TM7MD3	TM8MD3
03FAX																
03FBX	TMCKSEL1	TMCKSEL2	TMINSEL1	TMINSEL2										FBEWER		FEWSPD
03FCX						ANCTR0	ANCTR1	ANCTR2	ANBUF0	ANBUF1						
03FDX	IRQCNT	NF0CTR	NF1CTR	NF2CTR	NF3CTR	NF4CTR		LVLMD								
03FEX	Reserved	NMICR	IRQ0ICR	IRQ1ICR	IRQ2ICR	IRQ3ICR	IRQ4ICR	TS0DTICR	TS0DEICR	TS0CICR	TS0ATICR	TS1DTICR	TS1DEICR	TS1CICR	TS1ATICR	TM0ICR
03FFX	TM1ICR	TM2ICR	TM3ICR	TM6ICR	TBICR	TM7ICR	TM7OC2ICR	TM8ICR	TM8OC2ICR	PWMOVICR	PWMUDICR	SC0TICR	SC0RICR	ADICR	PERIILR	Reserved

Table:2.2.4 Register Map (MN101EFA7)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
03DAx	TS0TMD	TS0CKMD	TS0TCHSEL	TS0RESULT	TS0ERROR	TS0ADCNT										
03DBx	TS0CH0 EXDATL	TS0CH0 EXDATH	TS0CH1 EXDATL	TS0CH1 EXDATH	TS0CH2 EXDATL	TS0CH2 EXDATH	TS0CH3 EXDATL	TS0CH3 EXDATH	TS0CH4 EXDATL	TS0CH4 EXDATH	TS0CH5 EXDATL	TS0CH5 EXDATH	TS0CH6 EXDATL	TS0CH6 EXDATH	TS0CH7 EXDATL	TS0CH7 EXDATH
03DCx	TS0CH0 CTDATL	TS0CH0 CTDATH	TS0CH1 CTDATL	TS0CH1 CTDATH	TS0CH2 CTDATL	TS0CH2 CTDATH	TS0CH3 CTDATL	TS0CH3 CTDATH	TS0CH4 CTDATL	TS0CH4 CTDATH	TS0CH5 CTDATL	TS0CH5 CTDATH	TS0CH6 CTDATL	TS0CH6 CTDATH	TS0CH7 CTDATL	TS0CH7 CTDATH
03DDx																
03DEx																
03DFx																
03E0X	PWMMDL	PWMMDH	PWMSELL	PWMSELH	PWMSETL	PWMSETH	TCMPAL	TCMPAH	TCMPBL	TCMPBH	TCMPCL	TCMPCH	OUTMD	DTMSET	DTMSET1	
03E1X	PWMBCL	PWMBCH	BCSTR	PWMOFFL	PWMOFFH	IRQCULL	PWMTMCNT	RELCTR	PWMODR	RELSTAT	PWMCMP1	PWMCMP2				
03E2x																
03E3x																
03E4x																
03E5X	PRTKEY						OSCLOCK									
03E6X	TS0ATCNT0	TS0ATCNT1	TS0ATTRC1	TS0ATMAP0L	TS0ATMAP0M		TS0ATREGAP									
03E7X	P0OUT		P2OUT			P5OUT	P6OUT	P7OUT	P8OUT	P9OUT	PAOUT					
03E8X	P0IN		P2IN			P5IN	P6IN	P7IN	P8IN	P9IN	PAIN					
03E9X	P0DIR		P2DIR			P5DIR	P6DIR	P7DIR	P8DIR	P9DIR	PADIR					
03EAX	P0PLU		P2PLU			P5PLUD	P6PLU	P7PLU	P8PLU	P9PLUD	PAPLU					SELUD
03EBX	P0OMD1					P5OMD	P6OMD		P8OMD		PAOMD					SELUD2
03ECX	P0OMD2					P5IMD				P9IMD	PAIMD					Reserved
03EDX																
03EEX	LEDCNT															
03EFX	P0ODC					P5ODC	P6ODC	P7ODC							Reserved	Reserved
03F0X	CPUM	MEMCTR	WDCTR	DLYCTR	Reserved		HANDSHAKE	AUCTR		Reserved	SBNKR	DBNKR	Reserved	Reserved		Reserved
03F1X	SC0SEL	SC0MD0	SC0MD1	SC0MD2	SC0MD3	SC0STR	RXBUF0	TXBUF0		RSTFACT	OSCCNT	RCCNT	OSCSCNT		EDGDT	PLLCNT
03F2X	SC1SEL	SC1MD0	SC1MD1	SC1MD2	SC1MD3	SC1STR	RXBUF1	TXBUF1	SCINTSEL		Reserved	Reserved				
03F3X	SC2SEL	SC2MD0	SC2MD1	SC2MD2	SC2MD3	SC2STR	RXBUF2	TXBUF2						Reserved	KEYT3_1IMD	KEYT3_2IMD
03F4X															IRQEXPEN	IRQEXPDT
03F5X	SC4MD0	SC4MD1	SC4MD2	SC4MD3	SC4AD0	SC4AD1	SC4STR0	SC4STR1	RXBUF4	TXBUF4	SC4SEL				STB_EDG	STB_MSK
03F6X	TM0BC	TM1BC	TM0OC	TM1OC	TM0MD	TM1MD	CK0MD	CK1MD	TM2BC	TM3BC	TM2OC	TM3OC	TM2MD	TM3MD	CK2MD	CK3MD
03F7X		TMABC		TMAOC		TMAMD1		TMAMD2	TM6BC	TM6OC	TM6MD	TBCLR	TM6BEN		TM7MD4	TM8MD4
03F8X	TM7BCL	TM7BCH	TM7OC1L	TM7OC1H	TM7PR1L	TM7PR1H	TM7ICL	TM7ICH	TM7MD1	TM7MD2	TM7OC2L	TM7OC2H	TM7PR2L	TM7PR2H		
03F9X	TM8BCL	TM8BCH	TM8OC1L	TM8OC1H	TM8PR1L	TM8PR1H	TM8ICL	TM8ICH	TM8MD1	TM8MD2	TM8OC2L	TM8OC2H	TM8PR2L	TM8PR2H	TM7MD3	TM8MD3

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
03FAX																
03FBX	TMCKSEL1	TMCKSEL2	TMINSEL1	TMINSEL2										FBEWER		FEWSPD
03FCX						ANCTR0	ANCTR1	ANCTR2	ANBUF0	ANBUF1						
03FDX	IRQCNT	NF0CTR	NF1CTR	NF2CTR	NF3CTR	NF4CTR		LVLMD								
03FEX	Reserved	NMICR	IRQ0ICR	IRQ1ICR	IRQ2ICR	IRQ3ICR	IRQ4ICR	TS0DTICR	TS0DEICR	TS0CICR	TS0ATICR	Reserved	Reserved	Reserved	Reserved	TM0ICR
03FFX	TM1ICR	TM2ICR	TM3ICR	TM6ICR	TBICR	TM7ICR	TM7OC2ICR	TM8ICR	TM8OC2ICR	PWMOVICR	PWMUDICR	SC0TICR	SC0RICR	ADICR	PERIILR	Reserved

Table:2.2.5 Register Map (MN101EFA3)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
03DAx																
03DBx																
03DCx																
03DDx																
03DEx																
03DFx																
03E0X	PWMMDL	PWMMDH	PWMSELL	PWMSELH	PWMSETL	PWMSETH	TCMPAL	TCMPAH	TCMPBL	TCMPBH	TCMPCL	TCMPCH	OUTMD	DTMSET	DTMSET1	
03E1X	PWMBCL	PWMBCH	BCSTR	PWMOFFL	PWMOFFH	IRQCULL	PWMTMCNT	RELCTR	PWMODR	RELSTAT	PWMCMP1	PWMCMP2				
03E2x																
03E3x																
03E4x																
03E5X	PRTKEY						OSCLOCK									
03E6X																
03E7X	P0OUT		P2OUT	P3OUT	P4OUT	P5OUT	P6OUT	P7OUT	P8OUT	P9OUT	PAOUT	PBOUT				
03E8X	P0IN		P2IN	P3IN	P4IN	P5IN	P6IN	P7IN	P8IN	P9IN	PAIN	PBIN				
03E9X	P0DIR		P2DIR	P3DIR	P4DIR	P5DIR	P6DIR	P7DIR	P8DIR	P9DIR	PADIR	PBDIR				
03EAX	P0PLU		P2PLU	P3PLUD	P4PLU	P5PLU	P6PLU	P7PLU	P8PLU	P9PLUD	PAPLU	PBPLUD				SELUD
03EBX	P0OMD1					P5OMD	P6OMD		P8OMD		PAOMD					SELUD2
03ECX	P0OMD2			P3IMD						P9IMD	PAIMD	PBIMD				Reserved
03EDX																
03EEX	LEDCNT															
03EFX	P0ODC			P3ODC	P4ODC	P5ODC	P6ODC	P7ODC							Reserved	Reserved
03F0X	CPUM	MEMCTR	WDCTR	DLYCTR	Reserved		HANDSHAKE	AUCTR		Reserved	SBNKR	DBNKR	Reserved	Reserved		Reserved
03F1X	SC0SEL	SC0MD0	SC0MD1	SC0MD2	SC0MD3	SC0STR	RXBUF0	TXBUF0		RSTFACT	OSCCNT	RCCNT	OSCSCNT		EDGDT	PLLCNT
03F2X	SC1SEL	SC1MD0	SC1MD1	SC1MD2	SC1MD3	SC1STR	RXBUF1	TXBUF1	SCINTSEL		Reserved	Reserved				
03F3X	SC2SEL	SC2MD0	SC2MD1	SC2MD2	SC2MD3	SC2STR	RXBUF2	TXBUF2						Reserved	KEYT3_1IMD	KEYT3_2IMD
03F4X															IRQEXPEN	IRQEXPDT
03F5X	SC4MD0	SC4MD1	SC4MD2	SC4MD3	SC4AD0	SC4AD1	SC4STR0	SC4STR1	RXBUF4	TXBUF4	SC4SEL				STB_EDG	STB_MSK
03F6X	TM0BC	TM1BC	TM0OC	TM1OC	TM0MD	TM1MD	CK0MD	CK1MD	TM2BC	TM3BC	TM2OC	TM3OC	TM2MD	TM3MD	CK2MD	CK3MD
03F7X		TMABC		TMAOC		TMAMD1		TMAMD2	TM6BC	TM6OC	TM6MD	TBCLR	TM6BEN		TM7MD4	TM8MD4
03F8X	TM7BCL	TM7BCH	TM7OC1L	TM7OC1H	TM7PR1L	TM7PR1H	TM7ICL	TM7ICH	TM7MD1	TM7MD2	TM7OC2L	TM7OC2H	TM7PR2L	TM7PR2H		
03F9X	TM8BCL	TM8BCH	TM8OC1L	TM8OC1H	TM8PR1L	TM8PR1H	TM8ICL	TM8ICH	TM8MD1	TM8MD2	TM8OC2L	TM8OC2H	TM8PR2L	TM8PR2H	TM7MD3	TM8MD3
03FAX																

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
03FBX	TMCKSEL1	TMCKSEL2	TMINSEL1	TMINSEL2										FBEWER		FEWSPD
03FCX						ANCTR0	ANCTR1	ANCTR2	ANBUF0	ANBUF1						
03FDX	IRQCNT	NF0CTR	NF1CTR	NF2CTR	NF3CTR	NF4CTR		LVLMD								
03FEX	Reserved	NMICR	IRQ0ICR	IRQ1ICR	IRQ2ICR	IRQ3ICR	IRQ4ICR	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	TM0ICR
03FFX	TM1ICR	TM2ICR	TM3ICR	TM6ICR	TBICR	TM7ICR	TM7OC2ICR	TM8ICR	TM8OC2ICR	PWMOVICR	PWMUDICR	SC0TICR	SC0RICR	ADICR	PERIILR	Reserved

Table:2.2.6 Register Map (MN101EFA2)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
03DAx																
03DBx																
03DCx																
03DDx																
03DEx																
03DFx																
03E0X	PWMMDL	PWMMDH	PWMSELL	PWMSELH	PWMSETL	PWMSETH	TCMPAL	TCMPAH	TCMPBL	TCMPBH	TCMPCL	TCMPCH	OUTMD	DTMSET	DTMSET1	
03E1X	PWMBCL	PWMBCH	BCSTR	PWMOFFL	PWMOFFH	IRQCULL	PWMTMCNT	RELCTR	PWMODR	RELSTAT	PWMCMP1	PWMCMP2				
03E2x																
03E3x																
03E4x																
03E5X	PRTKEY						OSCLOCK									
03E6X																
03E7X	P0OUT		P2OUT			P5OUT	P6OUT	P7OUT	P8OUT	P9OUT	PAOUT					
03E8X	P0IN		P2IN			P5IN	P6IN	P7IN	P8IN	P9IN	PAIN					
03E9X	P0DIR		P2DIR			P5DIR	P6DIR	P7DIR	P8DIR	P9DIR	PADIR					
03EAX	P0PLU		P2PLU			P5PLUD	P6PLU	P7PLU	P8PLU	P9PLUD	PAPLU					SELUD
03EBX	P0OMD1					P5OMD	P6OMD		P8OMD		PAOMD					SELUD2
03ECX	P0OMD2					P5IMD				P9IMD	PAIMD					Reserved
03EDX																
03EEX	LEDCNT															
03EFX	P0ODC					P5ODC	P6ODC	P7ODC							Reserved	Reserved
03F0X	CPUM	MEMCTR	WDCTR	DLYCTR	Reserved		HANDSHAKE	AUCTR		Reserved	SBNKR	DBNKR	Reserved	Reserved		Reserved
03F1X	SC0SEL	SC0MD0	SC0MD1	SC0MD2	SC0MD3	SC0STR	RXBUF0	TXBUF0		RSTFACT	OSCCNT	RCCNT	OSCSCNT		EDGDT	PLLCNT
03F2X	SC1SEL	SC1MD0	SC1MD1	SC1MD2	SC1MD3	SC1STR	RXBUF1	TXBUF1	SCINTSEL		Reserved	Reserved				
03F3X	SC2SEL	SC2MD0	SC2MD1	SC2MD2	SC2MD3	SC2STR	RXBUF2	TXBUF2						Reserved	KEYT3_1IMD	KEYT3_2IMD
03F4X															IRQEXPEN	IRQEXPDT
03F5X	SC4MD0	SC4MD1	SC4MD2	SC4MD3	SC4AD0	SC4AD1	SC4STR0	SC4STR1	RXBUF4	TXBUF4	SC4SEL				STB_EDG	STB_MSK
03F6X	TM0BC	TM1BC	TM0OC	TM1OC	TM0MD	TM1MD	CK0MD	CK1MD	TM2BC	TM3BC	TM2OC	TM3OC	TM2MD	TM3MD	CK2MD	CK3MD
03F7X		TMABC		TMAOC		TMAMD1		TMAMD2	TM6BC	TM6OC	TM6MD	TBCLR	TM6BEN		TM7MD4	TM8MD4
03F8X	TM7BCL	TM7BCH	TM7OC1L	TM7OC1H	TM7PR1L	TM7PR1H	TM7ICL	TM7ICH	TM7MD1	TM7MD2	TM7OC2L	TM7OC2H	TM7PR2L	TM7PR2H		
03F9X	TM8BCL	TM8BCH	TM8OC1L	TM8OC1H	TM8PR1L	TM8PR1H	TM8ICL	TM8ICH	TM8MD1	TM8MD2	TM8OC2L	TM8OC2H	TM8PR2L	TM8PR2H	TM7MD3	TM8MD3
03FAX																

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
03FBX	TMCKSEL1	TMCKSEL2	TMINSEL1	TMINSEL2										FBEWER		FEWSPD
03FCX						ANCTR0	ANCTR1	ANCTR2	ANBUF0	ANBUF1						
03FDX	IRQCNT	NF0CTR	NF1CTR	NF2CTR	NF3CTR	NF4CTR		LVLMD								
03FEX	Reserved	NMICR	IRQ0ICR	IRQ1ICR	IRQ2ICR	IRQ3ICR	IRQ4ICR	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	TM0ICR
03FFX	TM1ICR	TM2ICR	TM3ICR	TM6ICR	TBICR	TM7ICR	TM7OC2ICR	TM8ICR	TM8OC2ICR	PWMOVICR	PWMUDICR	SC0TICR	SC0RICR	ADICR	PERIILR	Reserved

2.2.7 Flash Option

This LSI allocates memory area 0x040C1 to flash option area.
When turning on the power or restarting after reset, the CPU automatically reads the values set in the flash option area to set the timing to start a watchdog timer.
When using this LSI, be sure to set flash option.

Flash option in this LSI is allocated as below.

- Flash Option 1 (FLOP1: 0x040C1)

bp	7	6	5	4	3	2	1	0
Flag	WDEN_INIT	-	-	-	-	-	-	-

bp	Flag	Description
7	WDEN_INIT	Watchdog timer operation start timing selection 1: When the reset is released 0: “1” is written to WDEN flag
6 to 0	-	-

! For the specific functions of WDEN_INIT flag, refer to [Chapter 11 Watchdog Timer].

! As the flash option function is activated when the power is turned on or microcomputer restarts from reset, the data cannot be rewritten while the CPU is working.

2.3 Bus Interface

2.3.1 Bus Controller

The MN101E series provides separate buses to the internal memory and internal peripheral circuits to reduce bus line loads and thus realize faster operation.

There are three such buses: ROM bus, RAM bus, and peripheral expansion bus. They connect to the internal ROM, internal RAM, and internal peripheral circuits respectively. The bus control block controls the parallel operation of instruction read and data access. A functional block diagram of the bus controller is given below.

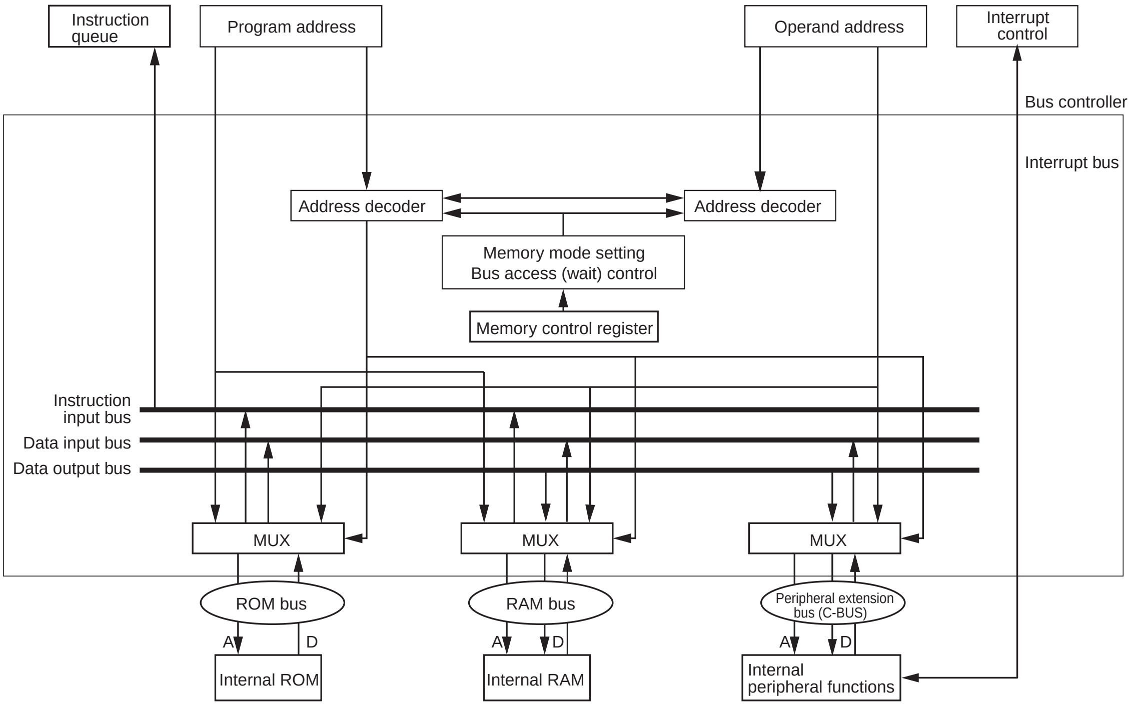


Figure:2.3.1 Functional Block Diagram of the Bus Controller

Memory control register (MEMCTR) can be used to set wait cycle to peripheral expansion bus (C-BUS) connected to internal peripheral circuits.

2.3.2 Control Registers

Bus interface is controlled by the memory control register (MEMCTR).

■ Memory Control Register (MEMCTR: 0x03F01)

bp	7	6	5	4	3	2	1	0
Flag	IOW1	IOW0	IVBM	Reserved	Reserved	IRWE	Reserved	Reserved
At reset	1	1	0	0	1	0	1	1
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7 to 6	IOW1 IOW0	Wait cycles when accessing special register area 00: No wait cycle 01: 1 wait cycle 10: 2 wait cycles 11: 3 wait cycles
5	IVBM	Base address specification for interrupt vector table 0:Interrupt vector base = 0x04000 1:Interrupt vector base = 0x00100
4	Reserved	Always set to “0”
3	Reserved	Always set to “1”
2	IRWE	Software write setting for interrupt request flag 0: Even if data is written to each interrupt control register (xxxICR), the state of the interrupt request flag (xxxIR) will not change. 1: Software write enable
1 to 0	Reserved	Always set to “11”

bp0, 1, 3 and 4 of the MEMCTR register are reserved. If accessing these flags, make sure to write the specified value.

The IOW1 to IOW0 wait settings affect accesses to the special registers located at the addresses 0x03000 to 0x03FFF. After reset, MEMCTR specifies the fixed wait cycle mode with three wait cycles. Wait setting of IOW is a function, which CPU supports for special use, for example, when special function register or I/O is expanded to external. For this LSI, wait cycle setting is not always necessary. Select "no-wait cycle" in this LSI.

Always set IRWE flag of memory control register (MEMCTR) to “0” except in writing IR with software.
If operating interrupt control register xxxICR with software in setting IRWE flag to “1”, the interrupt request flag which is set to “1” by interrupt source may be cleared to “0”.
For example, if giving the bit operation order to interrupt control register xxxICR, it executes bit operation to read-out 1 byte and writes it back.
When the interrupt source occurs between reading out and writing back, IR flag may be cleared to “0” by mistake and then the interrupt source is missing. If IRWE flag is set to “0”, the interrupt source will not be missing.

2.4 Extended Calculation Instruction

2.4.1 Overview

This LSI contains the functions of 16-bit x 16-bit and 32-bit/16-bit calculation in addition to the existing calculation functions which can be executed by MN101C series. The executable calculation and execution cycles are as follows:

Table:2.4.1 List of Extended Calculation Functions

Calculation	Instruction	Operation	Execution cycle	PSW Flag variation			
				VF	NF	CF	ZF
16-bit x 16-bit multiplication (unsigned)	MOV 1,(0x3F07) Extended calculation macro instruction MULWU	DW0 * DW1→{DW1, DW0}	13 cycles	0	√	0	√
16-bit x 16-bit multiplication (signed)	MOV 2,(0x3F07) Extended calculation macro instruction MULW	DW0 * DW1→{DW1, DW0}	13 cycles	0	√	0	√
32-bit/16-bit division (unsigned)	MOV 4,(0x3F07) Extended calculation macro instruction DIVWU	{DW1, DW0}/A0→DW0···DW1	14 cycles	√	√	0	√

√: flag varies.

VF: The value is 0 when the division result can be expressed with the unsigned 16-bit, and otherwise is 1. When zero divide is executed, the value is 1. When the value is 1, the calculation ends in 6 executing cycles.The value is always 0 in case of multiplication.

NF: The value is 1 when the MSB of multiplication/division result is 1. It is 0 when the MSB of the result is 0. However, when VF is 1, the value is undefined.

CF: Always 0.

ZF: The value is 1 when the multiplication/division result is 0, and otherwise is 0. However, when VF is 1, the value is undefined.

For details, refer to [Chapter 2 2.5 Extended Calculation Macro Instruction].

2.4.2 Extended Calculation Control Register

Extended calculation can be executed by setting the extended calculation control flag after setting the multiplier to DW0, the multiplicand to DW1, the divisor to A0, and the dividend to DW1 and DW0 respectively.

■ Expanded Calculation Control Register (AUCTR: 0x03F07)

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	-	AUDIVU	AUMUL	AUMULU
At reset	-	-	-	-	-	0	0	0
Access	-	-	-	-	-	W	W	W

bp	Flag	Description
7 to 3	-	-
2	AUDIVU	Unsigned division execution 0: Disabled 1: Enabled
1	AUMUL	Signed multiplication execution 0: Disabled 1: Enabled
0	AUMULU	Unsigned multiplication execution 0: Disabled 1: Enabled

When calculation is finished, each flag is cleared to "0" by hardware.

Do not set a number of bits at the same time.

2.4.3 Execution of Extended Calculation

■ Execution of 16-bit x 16-bit multiplication (unsigned)

1. Store the multiplier to DW0 register and the multiplicand to DW1 register.
2. Execute MOV 1, (0x03F07) (Extended calculation macro instruction MULWU).
3. The value of the unsigned 16-bit of DW0 register is multiplied by the unsigned 16-bit of DW1 register. Then the upper 16-bit of the result (32-bit) is stored in DW1 register and the lower 16-bit is stored in DW0 register.

■ Execution of 16-bit x 16-bit multiplication (signed)

1. Store the multiplier to DW0 register and the multiplicand to DW1 register.
2. Execute MOV 2, (0x03F07) (Extended calculation macro instruction MULW).
3. The value of the signed 16-bit of DW0 register is multiplied by the signed 16-bit of DW1 register. Then the upper 16-bit of the results (32-bit) is stored in DW1 register and the lower 16-bit register is stored in DW0 register.

■ Execution of 32-bit / 16-bit division (unsigned)

1. Store the upper 16-bit of the dividend to DW1 register, the lower 16-bit of the dividend to DW0 register, and the divisor to A0 register.
2. Execute MOV 4, (0x03F07) (Extended calculation macro instruction DIVWU).
3. The value of the unsigned 32-bit which is stored in the DW1 register (upper 16-bit) and DW0 register (lower 16-bit) is divided by the value of the unsigned 16-bit of A0 register. Then the quotient 16-bit of the result is stored in DW0 register and the residue 16-bit of the result is stored in DW1 register.

! VF is 1 when the division result cannot be expressed with the unsigned 16-bit or zero divide is executed.

! When VF is 1, the calculation ends in 6 execution cycles. In this case, the result is undefined.

2.5 Extended Calculation Macro Instruction

Extended calculation macro instruction can be generated by specifying machine dependence option -mmuldivw by compiler of MN101C/MN101E series.

2.5.1 About Extended Calculation Macro Instruction

■ About this Table

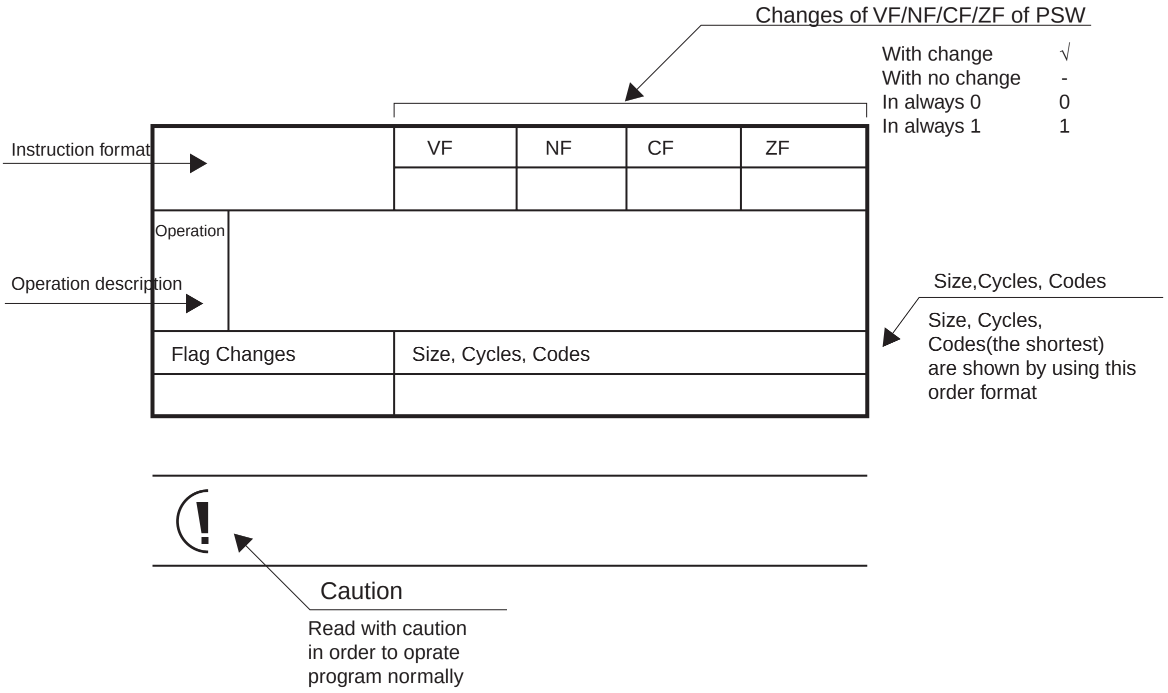


Figure:2.5.1 About this Table

■ Sign

*	Multiplication
/	Division
→	Substitution
...	Residue
{DW1,DW0}	32 bit data (high 16 bits in DW1 resister and low 16 bits in DW0 resister are stored)

2.5.2 MULWU

MULWU		VF	NF	CF	ZF
		0	√	0	√
Operation	DW0 *DW1 → {DW1, DW0} Multiplies the unsigned 16-bit value of DW0 register by the unsigned 16-bit value or DW1 register, and store the upper 16-bit of the result (32-bit) in the DW1 register and the lower 16-bit of the result in the DW0 register.				
Flag Changes		Size, Cycles, Codes			
VF: 0 NF: Set if the MSB of the result is 1, reset otherwise. CF: 0 ZF: Set if the result is 0, reset otherwise.		6 nibbles 13 cycles 0000 0010 0111 0000 0001 0000			

Store the multiplier to DW0 register and the multiplicand to DW1 register before executing this instruction.

This instruction is a macro instruction. The following operation is actually executed:

```
        MOV     1, (0x03F07); Set 1 to AUCTR register (address 0x03F07)
```

With this instruction, the address used in the handy addressing mode is updated to 0x03F07.

The updating example of handy addresses is as follows:

```
_CODE section code, pulic, 0

    mov     (0x100), d0
    mulwu
    mov     d0, (HA)        ; HA value is updated and
                            ; 0x03F07 address is accessed

    mov     (0x100), d0
    add     1, d0
    mov     d0, (HA)        ; access to HA value 0x100 address
```

2.5.3 MULW

<table>
<tr><td colspan="2" rowspan="2">MULW</td><td>VF</td><td>NF</td><td>CF</td><td>ZF</td></tr>
<tr><td>0</td><td>√</td><td>0</td><td>√</td></tr>
<tr><td>Operation</td><td colspan="5">DW0 *DW1 → {DW1, DW0}

Multiplies the unsigned 16-bit value of DW0 register by the unsigned 16-bit value or DW1 register, and store the upper 16-bit of the result (32-bit) in the DW1 register and the lower 16-bit of the result in the DW0 register.</td></tr>
<tr><td colspan="2">Flag Changes</td><td colspan="4">Size, Cycles, Codes</td></tr>
<tr><td colspan="2">VF: 0
NF: Set if the MSB of the result is 1, reset otherwise.
CF: 0
ZF: Set if the result is 0, reset otherwise.</td><td colspan="4">6 nibbles
13 cycles

0000 0010 0111 0000 0010 0000</td></tr>
</table>

Store the multiplier to DW0 register and the multiplicand to DW1 register before executing this instruction.

This instruction is a macro instruction. The following operation is actually executed:

```
        MOV     2, (0x03F07);set 2 to AUCTR register (address 0x03F07)
```

With this instruction, the address used in the handy addressing mode is updated to 0x03F07.

The updating example of handy addresses is as follows:

```
_CODE section code, pulic, 0

    mov     (0x100), d0
    mulwu
    mov     d0, (HA)        ; HA value is updated and
                            ; 0x03F07 address is accessed

    mov     (0x100), d0
    add     1, d0
    mov     d0, (HA)        ; access to HA value 0x100 address
```

2.5.4 DIVWU

<table>
<tr><td colspan="3" rowspan="2">DIVWU</td><td>VF</td><td>NF</td><td>CF</td><td>ZF</td></tr>
<tr><td>√</td><td>√</td><td>0</td><td>√</td></tr>
<tr><td>Operation</td><td colspan="6">{DW1, DW0}/A0 → DW0...DW1
Divides the unsigned 32-bit value which is stored in the DW1 register (upper 16-bit) and DW0 register (lower 16-bit) by the unsigned 16-bit value of A0 register, and stores the quotient 16-bit of the result in DW0 register and the residue 16-bit of the result in DW1 register.</td></tr>
<tr><td colspan="3">Flag Changes</td><td colspan="4">Size, Cycles, Codes</td></tr>
<tr><td colspan="2">If VF is 0
VF: 0 (if the quotient is an unsigned 16-bit value)
NF: Set if the MSB of the quotient is 1, reset otherwise.
CF: 0
ZF: Set if the MSB of the quotient is 0, reset otherwise.</td><td>If VF is 1
VF: 1 (if the quotient is not an unsigned 16-bit value)
NF: Undefined
CF: 0
ZF: Undefined</td><td colspan="4">6 nibbles
14 cycles
0000 0010 0111 0000 0100 0000</td></tr>
</table>

Set the upper 16-bit of the dividend to DW1 register, the lower 16-bit of the dividend to DW0 register, and the divisor to A0 register.

This instruction is a macro instruction. The following operation is actually executed:

MOV 4, (0x03F07); Set 4 to AUCTR register (address 0x03F07)

With this instruction, the address used in the handy addressing mode is updated to 0x03F07. The updating example of handy addresses is as follows:

```
_CODE section code, pulic, 0

    mov     (0x100), d0
    mulwu
    mov     d0, (HA)        ; HA value is updated and
                            ; 0x03F07 address is accessed

    mov     (0x100), d0
    add     1, d0
    mov     d0, (HA)        ; access to HA value 0x100 address
```

! VF is 1 when the division result cannot be expressed with the unsigned 16-bit or zero divide is executed.

! When VF is 1, the calculation ends in 6 execution cycles. In this case, the result is undefined.

2.6 Reset

2.6.1 Reset operation

The CPU contents are reset and registers are initialized when the NRST pin (P27) is pulled to low.

■ Initiating a Reset

There are two methods to initiate a reset.

1. Drive the NRST pin low.

   NRST pin should be held "low" for more than 1 μs.

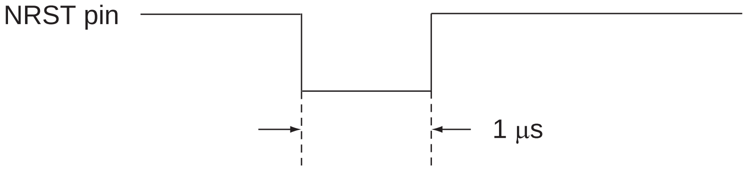


Figure:2.6.1 Minimum Reset Pulse Width

2. Setting the P2OUT7 flag of the P2OUT register to "0", transferring to reset by program (software reset) can be executed.

   If the internal LSI is reset and register is initiated, the P2OUT7 flag becomes "1" and reset is released.

! This LSI is activated in NORMAL mode (RC mode) in which the base clock is internal high frequency output.

! Refer to [Chapter 1 1.7.4 Power Supply] for reset control at power supply on.

! In this LSI, the internal oscillation is operated during reset.

! Pin NRST does not output “Low” level by software reset factors (watchdog reset and reset with programming P2OUT7 flag). The reset factor which occurred last time can be confirmed by reading RSTFACT register.

■ Reset Sequence

1. When reset pin comes to high level from low level, the internal binary counter (also used as watchdog timer) starts counting the system clock (fs). The length of time required from the start of counting to overflow is called "oscillation stabilization wait".

2. Internal registers and special function registers are initiated during the reset period.

Register	Address	R/W	Initial value	Description
PSW	-		0x00	Processor status word
PC	-		Address stored in 0x04000	Program counter
SP	-		undefined	Stack Pointer
An	-		undefined	Address register
Dn	-		undefined	Data register
CPUM	0x03F00	R/W	0x20	CPU mode control register
MEMCTR	0x03F01	R/W	0xCB	Memory control register
xxxICR	0x03FE2 to 0x03FFE	R/W	0x00	Maskable interrupt control register

3. Internal resets are cleared when the oscillation stabilization wait ends and program execution begins from the address written in the vector table at address 0x4000.

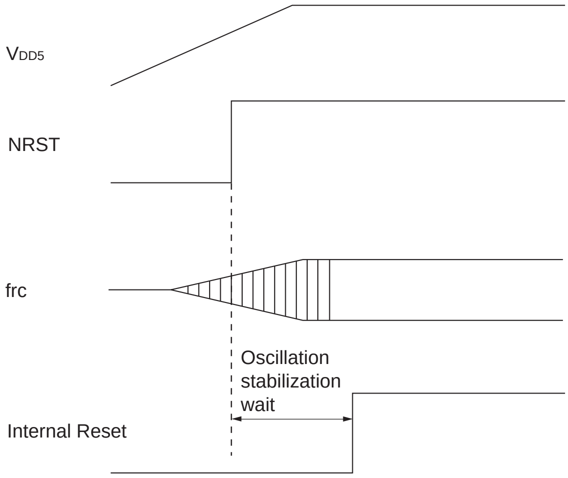


Figure:2.6.2 Reset Release Sequence

The value of internal RAM is uncertain at power-on because it is not initialized by reset.
The internal RAM needs to be initialized before used.

2.6.2 Oscillation Stabilization Wait time

The oscillation stabilization wait time is the period required for a stopped oscillation circuit to reach stable oscillation. An oscillation stabilization wait time is automatically inserted when a reset is cleared or when recovering from STOP mode. When recovering from STOP mode, the oscillation stabilization wait time control register (DLYCTR) is set to select the oscillation stabilization wait time. At releasing from reset, oscillation stabilization wait time is fixed.

The timer which counts oscillation stabilization wait time is also used as a watchdog timer except when a reset is cleared and recovering from STOP mode.

The watchdog timer is initiated at reset or in STOP mode, and starts counting from the initial value (0x0000) when system clock (fs) is selected as a clock source. After oscillation stabilization wait time, the timer continues counting as a watchdog timer if the overrun detection is enabled.

■ Oscillation Stabilization Wait (watchdog timer) Functions Block Diagram

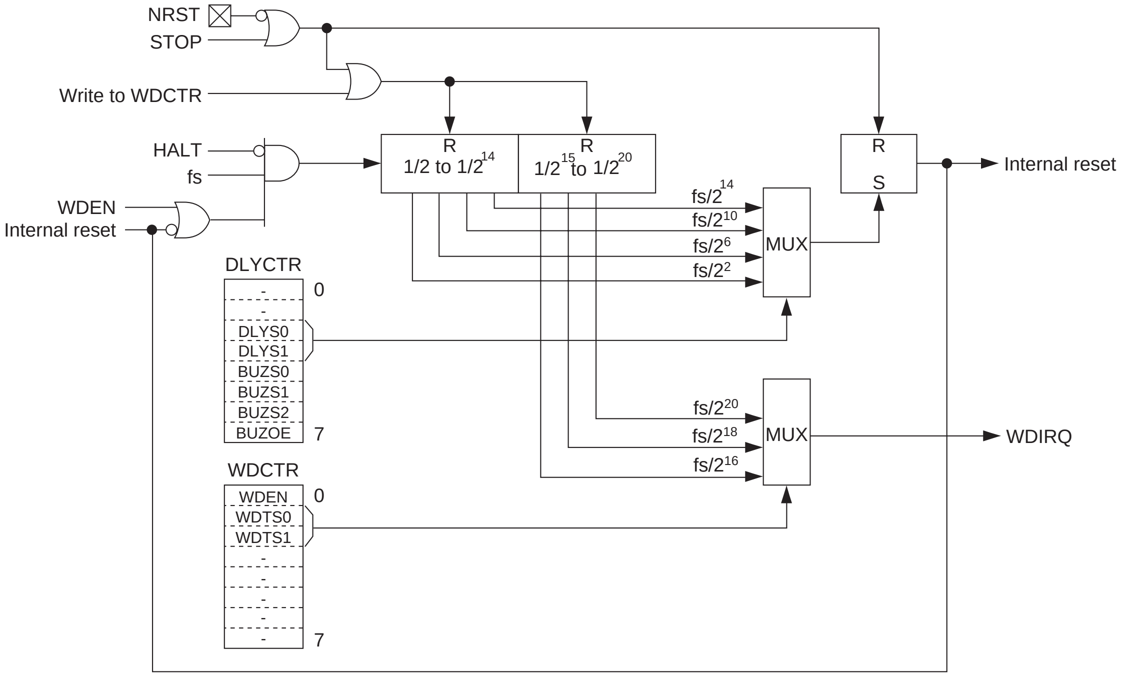


Figure:2.6.3 Block Diagram of Oscillation Stabilization Wait Time (Watchdog Timer)

■ Oscillation Stabilization Wait Control Register (DLYCTR: 0x03F03)

bp	7	6	5	4	3	2	1	0
Flag	BUZOE	BUZS2	BUZS1	BUZS0	DLYS1	DLYS0	-	-
At reset	0	0	0	0	0	1	-	-
Access	R/W	R/W	R/W	R/W	R/W	R/W	-	-

bp	Flag	Description
7	BUZOE	Output selection (Buzzer function) 0: Buzzer output disable 1: Buzzer output enable
6 to 4	BUZS2 BUZS1 BUZS0	Buzzer output frequency selection (Buzzer function) 000: fpll-div/2^{14} 001: fpll-div/2^{13} 010: fpll-div/2^{12} 011: fpll-div/2^{11} 100: fpll-div/2^{10} 101: fpll-div/2^{9} 110: fx/2^{4} 111: fx/2^{3}
3 to 2	DLYS1 DLYS0	Oscillation stabilization wait cycle selection 00: 2^{14} × System clock cycle 01: 2^{10} × System clock cycle 10: 2^{6} × System clock cycle 11: 2^{2} × System clock cycle
1 to 0	-	-

For the oscillation stabilization wait cycle required for high-speed/low-speed frequency, which is set by DLYS1 to DLYS0 flags, it is recommended to consult the oscillator manufacturer for determining appropriate values.

When returning from STOP mode, more than 100 μs of oscillation stabilization wait cycle must be set for internal regulator output stabilization wait.

Refer to [Chapter 12 Buzzer] for setup of bp7 to bp4 flags of oscillation stabilization wait time control register (DLYCTR).

■ Oscillation Stabilization Wait Time Control

When recovering from STOP mode, the oscillation stabilization wait time can be selected from 2^{14}, 2^{10}, 2^{6}, 2^{2} × system clock by setting the DLYS1 to DLYS0 flags of the oscillation stabilization wait control register (DLYCTR).

At reset release, the oscillation stabilization wait time is fixed to "2^{10} × system clock". System clock (fs) is determined by the CPU mode control register (CPUM).

Table:2.6.1 Oscillation Stabilization Wait Time

DLYS1	DLYS0	Oscillation stabilization wait time
0	0	2^{14} × System clock
0	1	2^{10} × System clock
1	0	2^{6} × System clock
1	1	2^{2} × System clock

For oscillation stabilization wait cycle, set the enough value to stabilize the oscillation circuit.

2.7 Auto Reset Functions

2.7.1 Overview

This LSI has one type of auto reset circuit to detect power supply voltage decrease.

The auto reset function whether using or not is selected by the ATRST pin.

- When using Auto reset --- ATRST pin: fixed at V_{DD5} level
- When not using auto reset--- ATRST pin: fixed at V_{SS} level

■ When power supply voltage is fallen to 0V.

When power supply voltage is fallen into the power supply change rate is upper 2 ms/V, Auto reset circuit outputs reset signal, if the power supply voltage becomes power detection level (V_{RST2}) or less.
After the internal reset reaction time (max: 400 µs) passes, LSI becomes a reset state.

■ When power supply voltage rises after detecting the power detection level (V_{RST2}) or less.

LSI can't become a reset state from detect power detection level (V_{RST2}) or less to the internal reset reaction time.
The power supply is risen after the internal reset reaction time passes, or the power supply change rate is lower 20 ms/V, because this state of LSI can't be guaranteed.
And, when the power supply (V_{DD5}) rises 4.0 V or less, observe [Chapter 1 1.7.4 Power Supply].

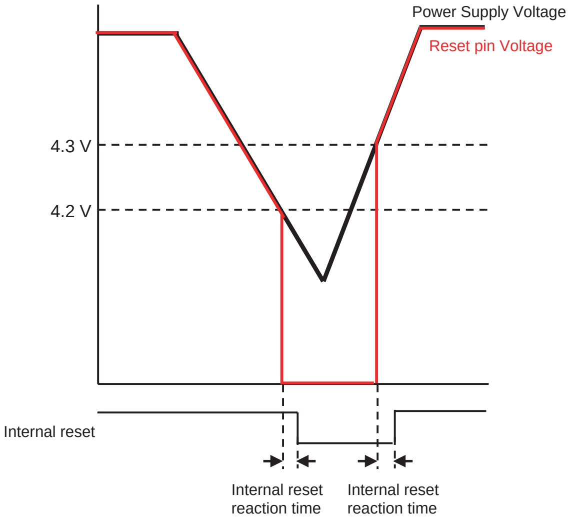


Figure:2.7.1 Power Supply and Reset Input Voltage

Refer to [Chapter 1 1.5.5 Auto Reset Characteristics] for V_{RST1} and V_{RST2}.

2.8 Reset Factor Determination

2.8.1 Overview

This LSI can detect reset factors (Hardware reset or Software reset).

- Hardware reset --- External reset (P27), Auto reset
- Software reset --- Watch dog reset and Reset with P2OUT7 register programming

The reset factors at the last reset can be determined by reading registers after releasing reset.

2.8.2 Registers

■ Reset Factor Determination Register (RSTFACT: 0x03F19)

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	-	-	-	RSTMON
At reset	-	-	-	-	-	-	-	0
Access	-	-	-	-	-	-	-	R

bp	Flag	Description
7 to 1	-	-
0	RSTMON	Reset Determination factor 0: Hardware reset occurs 1: Software reset occurs

If both hardware reset and software reset are detected at the same time, the RSTMON flag will be "0" (hardware reset).

Chapter 3　Clock Control

3.1 Overview

3.1.1 Overview

This LSI has following 2 types of clock generation circuit which supplies to CPU and peripheral circuits.

1) Internal high-speed oscillation circuit
Generates 16 MHz (Typ.) of clock inside a chip.

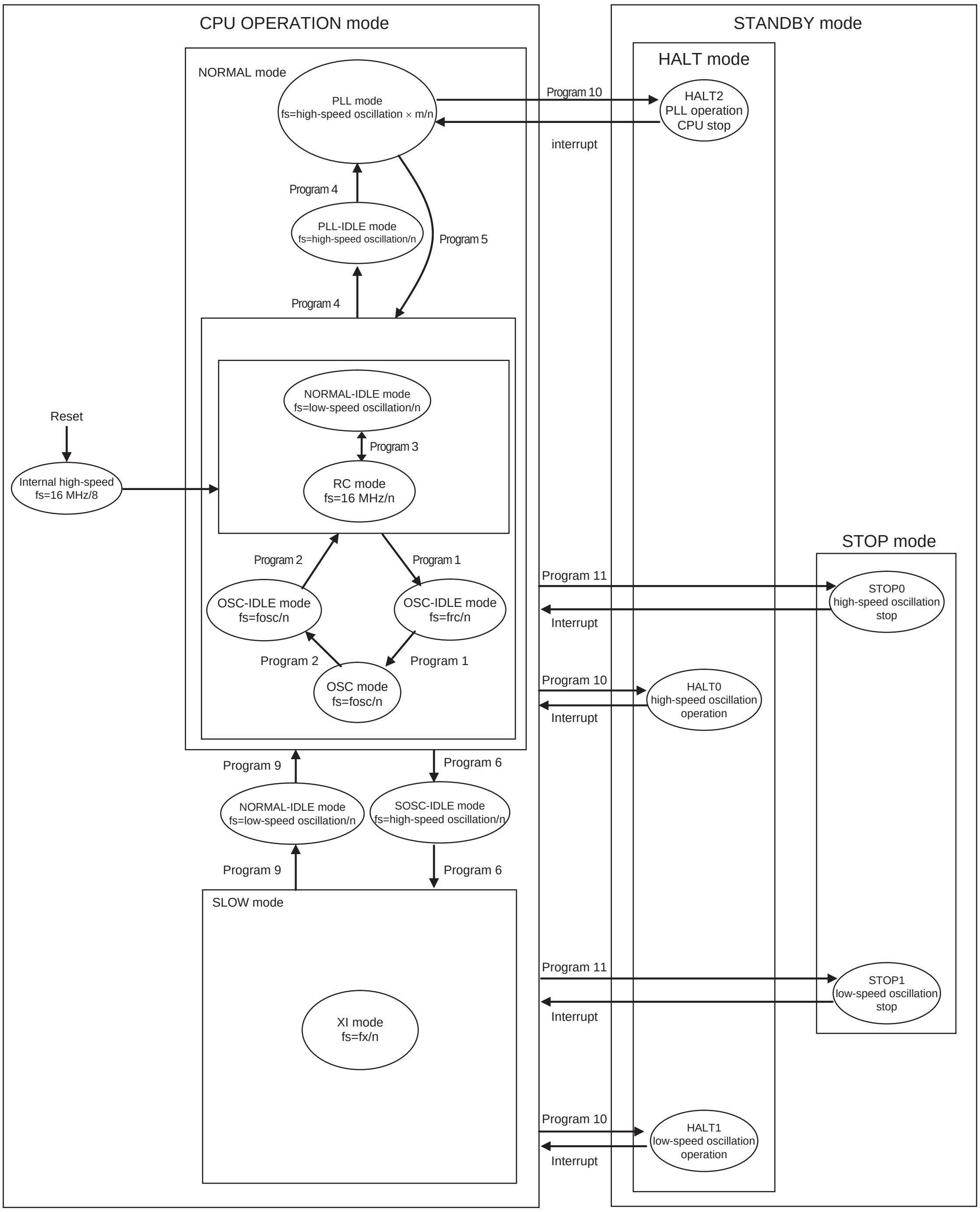
2) External high-speed oscillation circuit
Generates clocks by connecting crystal/ceramic oscillators to P25/OSC1 pin and P26/OSC2 pin.
At reset releasing, oscillation circuit is stopped.

3) External low-speed oscillation circuit:
This circuit generates clocks by connecting crystal/ceramic oscillators to P90/XI pin and P91/XO pin.
At reset releasing, oscillation circuit is stopped.

At PLL circuit multiplies clocks which generated in the internal high-speed oscillation circuit or the external high-speed oscillation circuit. At reset releasing, PLL circuit is stopped.

When using this PLL output clock as a system clock, the division ratio can be switched by the program.

At reset release, this product starts in RC mode (internal high-speed oscillation operation) fs=frc/8.



fs	: System clock
fosc	: External high-speed oscillation
frc	: Internal high-speed oscillation
fx	: External low-speed oscillation
m	: PLL Multiplication (set by PLLCNT register)
n	: Dividing ratio (set by CPUM register)

Figure:3.1.1 Transition Between Operation Modes

This LSI has 1 mode for CPU OPERATION mode (NORMAL and SLOW mode), and 2 modes for STANDBY mode (HALT mode and STOP mode).

■ NORMAL modes

There are three modes in NORMAL mode.

- RC mode
  In this mode, a clock which generated in the internal high-speed oscillation circuit is used as a system clock.
  At reset release, LSI starts with this mode (16 MHz, divided by 8).

- OSC mode
  In this mode, a clock which generated in the external high-speed oscillation circuit is used as a system clock.
  CPU enters this mode by controlling OSCCNT register.

- PLL mode
  In this mode, a clock which generated in the PLL circuit is used as a system clock.
  CPU enters this mode by controlling PLLCNT register.
  Which the internal high-speed oscillation and the external high-speed oscillation can be selected for an input clock to the PLL circuit. And the selection clock before transiting to PLL mode is used for an input clock.

■ SLOW mode

This mode executes the program using the low-speed operation clock. Low-power consumption while executing the program is possible because the high-speed oscillation circuit is stopped. There is one mode in SLOW mode.

- XI mode
  Int this mode, a clock generated in the external low-speed oscillation circuit is used as a system clock. CPU transfers to this mode by SLOW mode setting by the CPUM register after the SOSCCNT register is controlled.

■ HALT modes

In this mode, CPU stops but the oscillation circuit and PLL circuit are operating. An interrupt immediately returns CPU to operating mode. There are 2 modes in HALT mode.

- HALT0 mode
  The high-speed oscillation is operating. When an interrupt occurs, CPU enters RC/OSC mode.

- HALT1 mode
  Only the low-speed oscillation circuit operates, When an interrupt occurs, CPU transfers to SLOW mode.

- HALT2 mode
  The high-speed oscillation and PLL circuit are operating. When an interrupt occurs, CPU enters PLL mode.

■ STOP Mode

Both CPU and the oscillation circuits stops operating. The transition to the operation modes is performed after the operation of the oscillation circuit is started by an interrupt and wait the oscillation becomes stable.
There are 2 mode in STOP mode.

- STOP0 mode
  When an interrupt occurs, CPU enters NORMAL mode.

- STOP1 mode
  When an interrupt occurs, CPU enters SLOW mode.

■ IDLE Modes

There are 4 modes in IDLE mode.

- Normal-IDLE mode
  This mode is used to stabilize high-speed oscillation clock by programming, when transferring the CPU mode from SLOW to NORMAL, or changing the frequency of internal high-speed oscillation circuit.CPU enters this mode by controlling the CPUM register.

- OSC-IDLE mode
  This mode waits the program to stabilize the high-frequency oscillation clock when CPU enters from RC to OSC mode, and from OSC to RC mode. In this mode, the transition by CPUM register setting is not needed. The stabilization waiting is executed by the program in the status of operation mode before transition.

- PLL-IDLE mode
  This mode waits the program to stabilize the PLL clock when CPU enters from RC to OSC mode.
  In this mode, the transition by CPUM register setting is not needed.
  The stabilization waiting is executed by the program in the status of operation mode before transition.

- SOSC-IDLE mode
  This mode is used to stabilize the low-frequency oscillation clock by programming, when transferring the CPU mode from NORMAL to XI mode. In this mode, the transition set by CPUM register is not needed. The stabilization waiting is executed by the program in the status of operation mode before transition.

To reduce power consumption in STOP and HALT modes, it is necessary to check the stability of both the output current from pins and port level of input pins. For output pins, the output level should match the external level or direction control should be changed to input mode. For input pins, the external level should be fixed.

Transition to STANDBY mode is controlled by CPU mode control register (CPUM). Reset and interrupts are the return factors from STANDBY mode. A wait period is inserted for oscillation stabilization at reset and when returning from STOP mode, but not when returning from HALT mode.
Oscillation mode is automatically returned to the same state as existed before entering STANDBY mode.

In IDLE mode, the clock for the high-speed operation (fpll) is oscillated, however, do not operate the peripheral functions with fpll. Enable fpll operation of the peripheral functions after the high-speed operation state is changed to NORMAL mode.

3.1.2 Clock Control Function Block Diagram

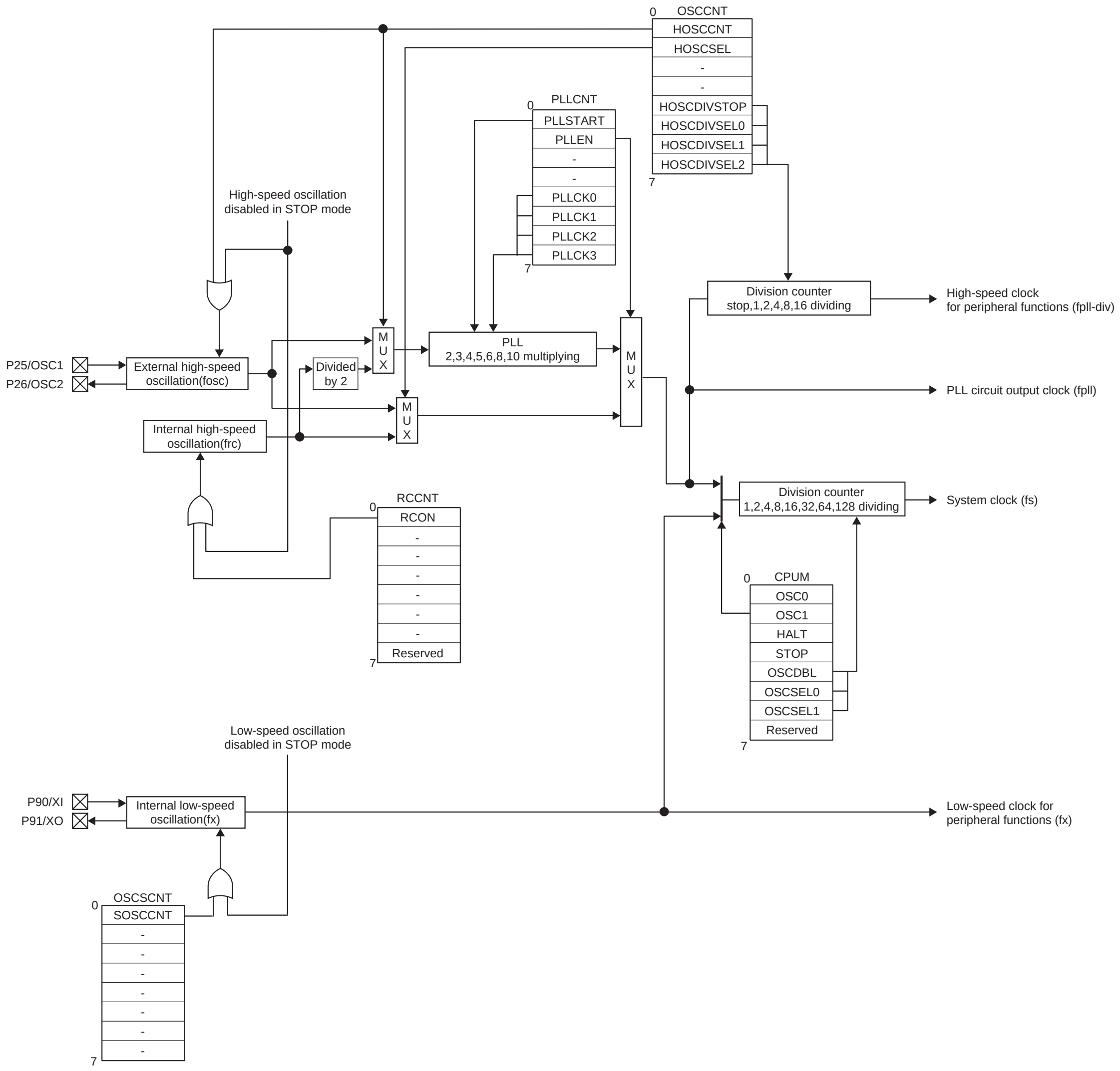


Figure:3.1.2 Clock Control Function Block Diagram

The clock which supplies to PLL circuit is selected by P25/P26 function selection flag HOSCCNT of the external high-speed oscillation control register (OSCCNT).
When the HOSCCNT is set to "1", this clock is used to supply to PLL circuit by the external high-speed oscillation.
And when the HOSCCNT is set to "0", this clock is used to supply to PLL circuit by the internal high-speed oscillation.
This clock is different from the clock selected by the operation clock selection flag (HOSCSEL) of the external high-speed oscillation control register (OSCCNT).

Table:3.1.1 Blocks which supply the high-speed clock for peripheral functions

Peripheral functions	PLL circuit output clock (fpll)	High-speed clock for peripheral functions (fpll-div)
8-bit timer	-	√
Simple 8-bit timer	-	√
16-bit timer	-	√
16-bit timer for motor control	-	√
Timer base timer	-	√
Free-running timer	-	√
Buzzer	-	√
Serial interface	-	√
Touch sensor timer	-	√

To set system clock (fs) > 10 MHz, set ROMHND flag of HANDSHAKE register in advance.

At microcontroller reset release, LSI starts with the following conditions:
Internal high-speed oscillation, fpll=frc (16 MHz), fs=fpll/8, Internal ROM access method = HANDSHAKE

3.2 Control Registers

3.2.1 Registers List

Table shows the registers to control the clock generation function.

Table:3.2.1 Clock Generation Function Control Registers List

Table remarks √: With function -: Without function

Registers	Address	R/W	Function	Page	MN101EF A8/A3	MN101EF A7/A2
OSCLOCK	0x03E56	R/W	Oscillation control register protect register	III-13	√	√
CPUM	0x03F00	R/W	CPU mode control register	III-16	√	√
HANDSHAKE	0x03F06	R/W	Internal ROM access method control register	III-18	√	√
OSCCNT	0x03F1A	R/W	External high-speed oscillation control register	III-9	√	√
RCCNT	0x03F1B	R/W	Internal high-speed oscillation control register	III-11	√	√
PLLCNT	0x03F1F	R/W	Clock multiplication circuit control register	III-14	√	√
OSCSCNT	0x03F1C	R/W	External low-speed oscillation control register	III-12	√	√
FEWSPD	0x03FBF	R/W	Internal Flash control register	III-20	√	√

R/W: Readable/Writable

3.2.2 Oscillation Control Registers

■ External High-speed Oscillation Control Register (OSCCNT: 0x03F1A)

This register is used to control the external high-speed oscillation, select between the external high-speed oscillation (fosc) and the internal high-speed oscillation (frc), control and divide the high-speed oscillation for peripheral functions.

bp	7	6	5	4	3	2	1	0
Flag	HOSCDIV SEL2	HOSCDIV SEL1	HOSCDIV SEL0	HOSCDIV STOP	-	-	HOSC SEL	HOSC CNT
At reset	0	0	0	0	-	-	0	0
Access	R/W	R/W	R/W	R/W	-	-	R/W	R/W

bp	Flag	Description
7 to 5	HOSCDIVSEL2 HOSCDIVSEL1 HOSCDIVSEL0	High-speed oscillation dividing selection for peripheral functions 000: No dividing 001: Divide by 2 010: Divide by 4 011: Divide by 8 1XX: Divide by 16
4	HOSCDIVSTOP	High-speed oscillation operating selection for peripheral functions 0: Enable 1: Disable
3 to 2	-	-
1	HOSCSEL	Operating clock selection 0: Internal high-speed oscillation 1: External high-speed oscillation
0	HOSCCNT *1	P25/P26 function selection 0: General-purpose port 1: High-speed oscillation pin

*1 Rewriting is enable when HOSCSEL flag is “0”.

HOSCSEL flag and HOSCCNT flag must be written in NORMAL mode.
The operation switching in SLOW/IDLE/PLL mode is not guaranteed.

When changing the operating clock from the internal high-speed oscillation to the external high-speed oscillation, wait for the oscillation to stable after setting HOSCCNT flag before switching the operating clock by setting HOSCSEL flag.

When changing the operation clock from the external high-speed oscillation to the internal high-speed oscillation, wait for 30 μs or longer after setting the RCON flag of RCCNT register to “0”. Then set the HOSCSEL flag of OSCCNT register to “0”.

The oscillation stabilization wait time of the external high-speed oscillation should be set after operates the oscillation matching on the board and consults the manufacturers.

The clock which supplies to PLL circuit is selected by P25/P26 function selection flag HOSCCNT of OSCCNT register.
When the HOSCCNT is set to "1", this clock is used to supply to PLL circuit by the external high-speed oscillation.
And when HOSCCNT is set to "0", this clock is used to supply to PLL circuit by the internal high-speed oscillation.
This clock is different from the clock selected by the operation clock selection flag HOSCSEL of OSCCNT register.

When the HOSCCNT flag is cleared to set P25/P26 to general port after the operation clock is changed from the external high-speed oscillation to the internal high-speed oscillation, insert two NOP instructions and set the HOSCCNT flag.

Peripheral function operations must be stopped before changing the division setting and the operation setting of the high-speed oscillation for peripheral functions

Do not set the HOSCCNT flag to “0” when the external high-speed oscillation is selected by OSCSEL flag as the operation clock.

In OSCCNT register, the writing processing can be masked by setting of the LOCKEN flag of OSCLOCK register.
By this operation, unexpected register writing can be restricted.

■ Internal High-speed Oscillation Control Register (RCCNT: 0x03F1B)

This register is used to control operations of the internal high-speed oscillation.

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	-	-	Reserved	RCON
At reset	-	-	-	-	-	-	1	0
Access	-	-	-	-	-	-	R/W	R/W

bp	Flag	Description
7 to 2	-	-
1	Reserved	Always set to “1”.
0	RCON *1	Internal high-speed oscillation control 0: Enabled 1: Disabled

*1 Rewriting is enable when HOSCSEL flag is “0”.

RCON flag must be written in NORMAL mode.
The operation switching in SLOW/IDLE/PLL mode is not guaranteed.

When the RCON flag is set to stop the internal high-speed oscillation after the operation clock is changed to the external high-speed oscillation, set it after inserting two NOP instructions.

Do not set the RCON flag to "1" when the HOSCSEL flag of OSCCNT register is set to "0", and the operating clock is selected as the internal high-speed oscillation.

When changing the operating clock from the external high-speed oscillation to the internal high-speed oscillation, wait for 30 μs or longer after setting the RCON flag of RCCNT register to “0”. Then set the HOSCSEL flag of OSCCNT register to “0”.

When setting the HOSCSEL flag of OSCCNT register to change the operation clock from the internal high-speed oscillation to the external high-speed oscillation, the internal high-speed oscillation is disabled by setting the RCON flag. And the operating current can be reduced.

In RCCNT register, the writing processing can be masked by setting of the LOCKEN flag of OSCLOCK register.
By this operation, unexpected register writing can be restricted.

■ External Low-speed Oscillation Control Register (OSCSCNT: 0x03F1C)

This register is used to control external low-speed oscillation.

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	-	-	-	SOSCCNT
At reset	-	-	-	-	-	-	-	0
Access	-	-	-	-	-	-	-	R/W

bp	Flag	Description
7 to 1	-	-
0	SOSCCNT	P90/P91 function selection 0 : General port 1 : Low-speed oscillation pin

SOSCCNT flag must be written in NORMAL mode.
The operation switching in SLOW/IDLE/PLL mode is not guaranteed.

The oscillation stabilization wait time of the external low-speed oscillation should be set after operates the oscillation matching on the board and consults the manufacturer.

As OSCSCNT register is masked by setting the LOCKEN flag of OSCLOK register, unintentional register write can be prevented.

■ Oscillation Control Register Protect Register (OSCLOCK: 0x03E56)

This register is used to control writing to OSCCNT register and RCCNT register.

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	-	-	-	LOCKEN
At reset	-	-	-	-	-	-	-	0
Access	-	-	-	-	-	-	-	R/W

bp	Flag	Description
7 to 1	-	-
0	LOCKEN	Oscillation control register writing control 0: Enabled 1: Disabled

When the LOCKEN flag is set to "1", writing to OSCCNT, RCCNT and OSCSCNT registers is masked. But when reading, the setting values are read-out.

■ Clock Multiplication Circuit Control Register (PLLCNT: 0x03F1F)

This register is used to control the clock generation of 2 to 10 multiplications by using the oscillation selected by OSCCNT register.

bp	7	6	5	4	3	2	1	0
Flag	PLLCK3	PLLCK2	PLLCK1	PLLCK0	-	-	PLLEN	PLLSTART
At reset	0	0	0	0	-	-	0	0
Access	R/W	R/W	R/W	R/W	-	-	R/W	R/W

bp	Flag	Description
7 to 4	PLLCK3 PLLCK2 PLLCK1 PLLCK0	Multiplication setting 0000: Multiply by 2 (input frequency 4 to 7.5 MHz) 0001: Multiply by 2 (input frequency 7.5 to 10 MHz) 0010: Multiply by 3 (input frequency 4 to 5 MHz) 0011: Multiply by 4 (input frequency 4 to 7.5 MHz) 0100: Multiply by 4 (input frequency 7.5 to 10 MHz) 0101: Multiply by 5 (input frequency 4 to 6 MHz) 0110: Multiply by 5 (input frequency 6 to 8 MHz) 0111: Multiply by 6 (input frequency 4 to 5 MHz) 1000: Multiply by 8 (input frequency 4 to 5 MHz) 1001: Multiply by 10 (input frequency 4 MHz) 1010: Setting prohibited 1011: Setting prohibited 1100: Setting prohibited 1101: Setting prohibited 1110: Setting prohibited 1111: Setting prohibited
3 to 2	-	-
1	PLLEN	PLL clock enable 0: High-speed oscillation clock (fosc/frc) operation 1: PLL clock operation
0	PLLSTART	PLL operation control 0: PLL stop 1: PLL operation

Switch the operation clock to PLL output after setting the PLLSTART flag and wait for 100 μs or longer. Then set the PLLEN flag and switch the operation clock.

Do not change the multiplication setting by changing the PLLCK3 to PLLCK0 flags on PLL operation. Change the multiplication setting when the PLLSTART flag is "0".

Do not switch the PLL input clock by changing the HOSCCNT flag of OSCCNT register.
Make sure to change the PLL input clock when the PLLSTART flag is "0".

When stopping the PLL circuit by clearing the PLLSTART flag after the transition from PLL mode to RC mode or OSC mode, make sure to insert more than two NOP instructions.

The clock which supplies to PLL circuit is selected by P25/P26 function selection flag HOSCCNT of OSCCNT register.
When the HOSCCNT flag is set to "1", this clock is used to supply to PLL circuit by the external high-speed oscillation.
And when the HOSCCNT flag is set to "0", this clock is used to supply to PLL circuit by the internal high-speed oscillation.
This clock is different from the clock selected by the operation clock selection flag HOSCSEL of OSCCNT register.

The operation that the system clock proceeds 20 MHz is not guaranteed.
The division setting must be operated by CPU mode control register before transition to PLL mode. And transit to PLL mode before the system clock is set to 20 MHz or less.

■ CPU Mode Control Register (CPUM: 0x03F00)

This register is used to control the operation mode and division ratio of system clock.

bp	7	6	5	4	3	2	1	0
Flag	Reserved	OSCSEL1	OSCSEL0	OSCDBL	STOP	HALT	OSC1	OSC0
At reset	0	0	1	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7	Reserved	Always set to “0”.
6 to 5	OSCSEL1 OSCSEL0	Division ratio 00: 1 01: 4 10: 16 11: 64
4	OSCDBL	Internal system clock (fs) 0: Normal (fpll/2) 1: Double-speed (fpll)
3 to 0	STOP HALT OSC1 OSC0	CPU mode selection 0000: NORMAL 0001: NORMAL-IDLE 0011: SLOW 0100: HALT0/HALT2 0111: HALT1 1000: STOP0 1011: STOP1 Others: Setting prohibited

Do not change both settings of the clock switching function (OSCDBL, OSCSEL1 to 0 flags) and the STANDBY function (STOP, HALT, OSC1 and OSC0 flags) at the same time.

Table:3.2.2 Division Ratio Setting by Combination of OSCSEL and OSCDBL

OSCSEL1	OSCSEL0	OSCDBL	Division ratio
0	0	0	2
0	0	1	1
0	1	0	8
0	1	1	4
1	0	0	32
1	0	1	16
1	1	0	128
1	1	1	64

Set the division ratio of the oscillation clock and the operating mode transition within the range satisfying LSI operating condition and operating speed.
Refer to [Chapter 1 1.5 Electrical Characteristics].
Especially, when using an external high-speed oscillation clock as a system clock, note the relation among dividing ratios, the frequency of fosc and the maximum frequency of fs.

OSCDBL, OSCSEL1 and OSCSEL0 flags can be changed simultaneously.

Table:3.2.3 Operating Mode Control and Clock Oscillation/Halt

Operating mode	STOP	HALT	OSC1	OSC0	Status			
					OSC/RC	XI	System clock	CPU
NORMAL	0	0	0	0	Oscillate	Oscillate	OSC/RC/PLL	Operate
NORMAL-IDLE	0	0	0	1	Oscillate	Oscillate	XI	Operate
SLOW	0	0	1	1	Stop	Oscillate	XI	Operate
HALT0/HALT2	0	1	0	0	Oscillate	Oscillate	OSC/RC/PLL	Halt
HALT1	0	1	1	1	Stop	Oscillate	XI	Halt
STOP0	1	0	0	0	Stop	Stop	Stop	Stop
STOP1	1	0	1	1	Stop	Stop	Stop	Stop

The procedure for transition from NORMAL to HALT or STOP mode is shown below.

1. Clear the interrupt request flag (xxxIR) in the maskable interrupt control register (xxxICR) and set the interrupt enable flag (xxxIE) for the return factor.
2. To return using an maskable interrupt, set the MIE flag of PSW to "1" and set the interrupt mask (IM) to a level permitting the interrupt to be accepted.
3. Set CPUM register to HALT or STOP mode.

Set the IRWE flag of memory control register (MEMCTR) to clear the interrupt request flag by software. Be sure to clear the IRWE flag after the interrupt request flag is cleared.

■ Internal ROM Access Method Control Register (HANDSHAKE: 0x03F06)

This register is used to control the access method for the internal ROM area. In this LSI, the access method for the internal ROM area should be set by relation between power supply voltage (V_{DD5}) and system clock frequency (fs).

Table:3.2.4 ROMHND Flag Setting List

Power supply voltage (V_{DD5})	System clock (fs)	ROMHND flag
$4.0\ V \leq V_{DD5} \leq 5.5V$	$fs \leq 10$ MHz	0
	10 MHz $< fs \leq 20$ MHz	1

When HANDSHAKE is set to the internal ROM access method, access cycles increases at the conditions of below table.

Table:3.2.5 Extension Conditions of Access Cycle at HANDSHAKE access

Previous conditions	Conditions of access extension	Cycles
Reading instructions whose branch destination is located in odd address	Reading branch instructions located in consecutive even address	2
Reading branch instructions located in odd address	Reading instructions whose branch destination is located in odd address	3
Reading branch instructions located in even address	Reading instructions whose branch destination is located in odd address	2
Reading branch instructions located in odd address	Reading instructions whose branch destination is located in even address	3
Reading branch instructions located in even address	Reading instructions whose branch destination is located in even address	2
Reading instructions located in odd address	Data access to odd address	3
Reading instructions located in even address	Data access to odd address	2
Reading instructions located in odd address	Data access to even address	3
Reading instructions located in even address	Data access to even address	2

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	-	ROMHND	-	-
At reset	-	-	-	-	-	1	-	-
Access	-	-	-	-	-	R/W	-	-

bp	Flag	Description
7 to 3	-	-
2	ROMHND	Internal ROM access method selection 0: Normal access 1: HANDSHAKE access
1 to 0	-	-

Set the ROMHND flag in advance to select HANDSHAKE access when transition to the power supply voltage and the system clock frequency of the ROMHND flag settings described in Table:3.2.4

Clear the ROMHND flag and select normal access when operating with the power supply voltage and the system clock frequency of the ROMHND flag clearing described in Table:3.2.4. The access timing to the internal ROM area improves.

At reset release, LSI starts with the internal ROM access method = HANDSHAKE.

On the condition of fs ≤ 10 MHz (4.0 V ≤ V_{DD5} ≤ 5.5 V), CPU processing performance in HANDSHAKE access (ROMHND = 1) is inferior to that in NORMAL access (ROMHND = 0).

■ Internal Flash Control Register (FEWSPD: 0x03FBF)

This LSI is equipped with flash memory as internal ROM.
When an access to a ROM area is not required since a program is executed on RAM, the current to operate the internal flash can be reduced by stopping the internal flash memory by FEWSPD register.

bp	7	6	5	4	3	2	1	0
Flag	Reserved	-	-	-	NSTOP	-	-	Reserved
At reset	0	-	-	-	0	-	-	1
Access	R/W	-	-	-	R/W	-	-	R

bp	Flag	Description
7	Reserved	Always set to “0”.
6 to 4	-	-
3	NSTOP	Internal flash memory operate/stop 0: Operating 1: Stop
2 to 1	-	-
0	Reserved	-

Make sure to set the NSTOP flag while the program is executed on the RAM. When NSTOP flag is set to “1” while ROM is executed, access to internal flash memory is disabled. Errors may occur in CPU.

After the NSTOP flag is cleared, switch to the program on the internal flash after waiting for 20 μs or longer.

BSET/BCLR instruction must be used for setting of NSTOP flag.

In-circuit emulator cannot stop internal flash memory.
Debug the function of selecting operations (operating/stop) of internal flash memory under on-board debugging environment.

3.3 Transition to Each Mode

The sample programs for transition to each mode is shown below.

3.3.1 NORMAL Mode

■ Transition from RC mode to OSC mode

Transition from RC mode to OSC mode requires passing through IDLE mode. In OSC-IDLE mode, CPU operates with the internal high-speed oscillation clock before transition to the modes.

The frequency used as a system clock varies depending on the power supply voltage.

```
Program1
            BSET (OSCCNT)0        ; Set P25/P26 to the high-speed oscillation pin
            MOV x'FF', D0
LOOP        ADD -1, D0            ; Loop to wait 76 µs at 20 MHz operation
            BNE LOOP
            BSET (OSCCNT)1        ; Set OSC mode
            NOP
            NOP
            BSET (RCCNT)0         ; Stop the internal high-speed oscillation circuit
```

The oscillation stabilization wait time of the external high-speed oscillation should be set after operating the oscillation matching on the board and consulting with other manufacturers.

When the internal high-speed oscillation is stopped to set the RCON flag after the operating clock is changed to the external high-speed oscillation, insert two NOP instructions and set the RCON flag.

When setting the HOSCSEL flag of OSCCNT register to change the operation clock from the internal high-speed oscillation to the external high-speed oscillation, the internal high-speed oscillation is disabled by setting the RCON flag. As a result, the operating current can be reduced.

■ Transition from OSC mode to RC mode

Transition from RC mode to OSC mode requires passing through IDLE mode. In OSC-IDLE mode, CPU operates with the internal high-speed oscillation clock before transition to the modes.

The frequency used as a system clock varies depending on the power supply voltage.

```
Program2
                BCLR (RCCNT)0          ; Operate the internal high-speed oscillation circuit
                MOV x'46', D0
                BSET (HANDSHAKE)2      ; Set the internal ROM access method to HANDSHAKE
                BCLR (OSCCNT)1         ; Set RC mode
                NOP
                NOP
                BCLR (OSCCNT)0         ; Set P25/P26 pin to general pin
```

When changing the operation clock from the external high-speed oscillation to the internal high-speed oscillation, wait more than 30 μs after setting the RCON flag of the RCCNT register to “0”. Then set the HOSCSEL flag of the OSCCNT register to “0”.

When the HOSCCNT flag is cleared to set P25/P26 to a general port after the operation clock is changed from the external high-speed oscillation to the internal high-speed oscillation, insert two NOP instructions and set the HOSCCNT flag.

Set the ROMHND flag in advance to select HANDSHAKE access when transition to the power supply voltage and the system clock frequency of the ROMHND flag settings described in Table:3.2.4.

■ Transition from RC mode or OSC mode to PLL mode

Transition to PLL mode requires passing through PLL-IDLE mode. In PLL-IDLE mode, CPU operates with the high-speed oscillation clock (RC, OSC).

The frequency used as a system clock varies depending on the power supply voltage.

```
Program3
            MOV x'40', (PLLCNT)     ; Set to 4 multiplication
            BSET (PLLCNT)0          ; Operate PLL circuit
            MOV x'A6', D0
LOOP        ADD -1, D0              ; Loop to wait 100 µs at 10 MHz
            BNE LOOP
            BSET (HANDSHAKE)2       ; Set the internal ROM access method to HANDSHAKE
            BSET (PLLCNT)1          ; Set PLL mode
```

When changing the operation clock to PLL output, set the PLLSTART flag and wait 100 µs or longer. Then set the PLLEN flag.

Set the ROMHND flag in advance to select HANDSHAKE access when transition to the conditions described in Table:3.2.4.

The clock which supplies to PLL circuit is selected by P25/P26 function selection flag HOSCCNT of the external high-speed oscillation control register (OSCCNT).
When the HOSCCNT is set to "1", this clock is used to supply to PLL circuit by the external high-speed oscillation.
And when the HOSCCNT is set to "0", this clock is used to supply to PLL circuit by the internal high-speed oscillation.
This clock is different from the clock selected by the operation clock selection flag (HOSCSEL) of the external high-speed oscillation control register (OSCCNT).

■ Transition from PLL mode to RC mode or OSC mode

The transition from PLL mode to RC mode or OSC mode can be operated by writing to PLLCNT register.
In this case, transition through IDLE state is not needed.

```
Program4
             BCLR (PLLCNT)1          ; Set RC mode or OSC mode
             NOP
             NOP
             BCLR (PLLCNT)0          ; Stop PLL circuit
```

! When stopping the PLL circuit by clearing the PLLSTART flag after the transition from PLL mode to RC mode or OSC mode, make sure to insert more than two NOP instructions.

3.3.2 SLOW Mode

Transition from NORMAL to SLOW mode, when the low-speed clock (XI) has fully stabilized, can be done by writing to the CPU mode control register.
In this case, transition through IDLE state is not needed.

■ Transition from NORMAL mode to XI mode

```
Program 5
            BSET (OSCSCNT)0        ; Set P90/P91 to the low-speed oscillation pin.
            MOV x'FF', D0
            MOVW x'0F42', A0
LOOP2
LOOP1       ADD -1, D0             ; Loop to wait 300 ms at 20 MHz
            BNE LOOP1
            ADDW -1, A0
            BNE LOOP2
            MOV x'03', D0          ; Set SLOW mode.
            MOV D0, (CPUM)
```

The oscillation stabilization wait time of the external low-speed oscillation should be set after operates the oscillation matching on the board and consults the manufacture of your oscillator.

After the reset is released, wait 30 ms or longer to change the operating mode to SLOW, HALT1 or STOP.

■ Disable the Internal Flash in SLOW mode

In SLOW mode, the power consumption can be reduced by executing the program on the RAM to disable the internal flash.

```
Program 6
        Transfer required program to internal RAM area
        MOV x'30', D0               ; Set MIE to 0 and disable all maskable interrupts.
        MOV D0, PSW
        BSET (MEMCTR)5              ; Change the interrupt vector table base address on the RAM.
        Branch to the internal RAM area
        BSET (FEWSPD)3              ; Disable the internal flash.
        MOV x'70', D0               ; Set MIE to 1 and enable all maskable interrupts.
        MOV D0, PSW
```

■ Enable the Internal Flash in SLOW mode

```
Program 7
        MOV x'30',D0                ; Set MIE to 0 and disable all maskable interrupts.
        MOV D0, PSW
        BCLR (FEWSPD)3              ; Enable the internal flash.
        Branch to the internal ROM area
        MOV x'70', D0               ; Set MIE to 1 and enable all maskable interrupts.
        MOV D0, PSW
```

Make sure to set the NSTOP flag while the program is executed on the RAM.
When the NSTOP flag is set to "1" and executing ROM, errors occurs in CPU because an access to the internal flash can not be operated.

After the NSTOP flag is cleared, switch to the program on the internal flash after waiting for 20 µs or longer.

BSET/BCLR instruction must be used for setting NSTOP flag.

■ Transition from SLOW mode to NORMAL mode

For transition from SLOW to NORMAL mode, the program must be hold at the IDLE state until the high-speed clock oscillation is fully stable. In IDLE mode, CPU operates with the low-speed clock.

```
Program 8
            MOV x'01', (CPUM)       ; Set NORMAL-IDLE mode.
            MOV x'01', D0
LOOP        ADD -1, D0              ; Loop to wait 200 µs at 32 kHz
            BNE LOOP
            MOV x'00', (CPUM)       ; Set NORMAL mode.
```

When changing from SLOW mode to RC mode, make sure to wait 30 µs or longer at NORMAL-IDLE mode before switch to NORMAL mode.

When changing from SLOW to OSC mode, make sure to wait until the high-speed clock oscillation is fully stabilize at NORMAL-IDLE mode before switch to NORMAL mode.
The oscillation stabilization wait time of the external high-speed oscillation should be set after operates the oscillation matching on the board and consults the manufacturers.

3.3.3 STANDBY Mode

The program operates the transition from CPU OPERATION mode to STANDBY mode. Interrupts operate the return to CPU OPERATION mode.

Before transition to STANDBY mode, the following settings are required.

1. Set the maskable interrupt enable flag (xxxIE) and the maskable interrupt control register (xxxICR) of the processor status word (PSW) to "0" to disable all maskable interrupts temporarily.
2. Specify an interrupt factor which enables recovering from STANDBY mode to CPU OPERATION mode, and set the interrupt enable flag (xxxIE) of the maskable interrupt control register (xxxICR) only. Set MIE of PSW.

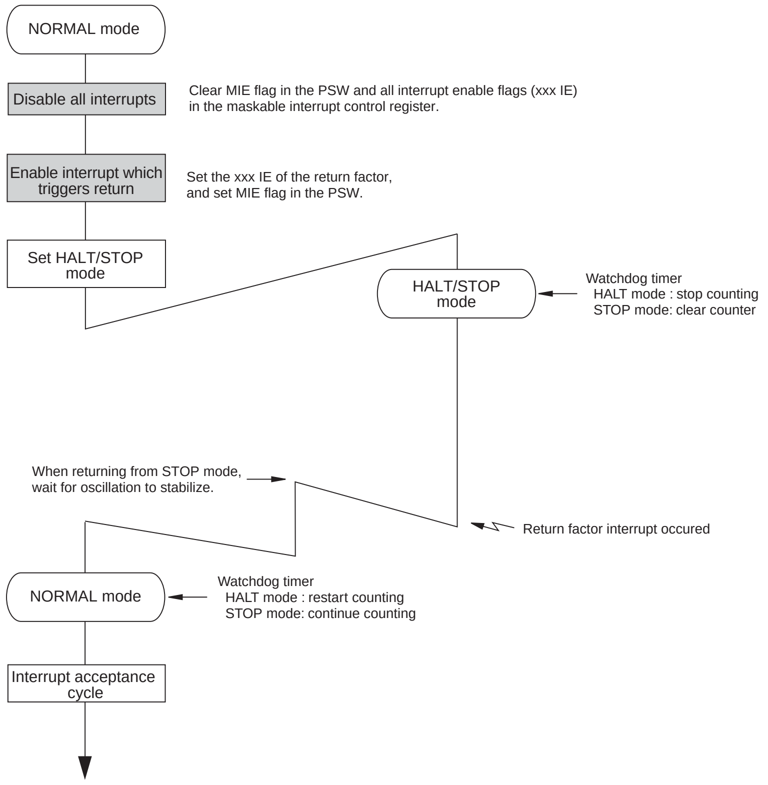

NORMAL mode
Disable all interrupts
Clear MIE flag in the PSW and all interrupt enable flags (xxx IE) in the maskable interrupt control register.
Enable interrupt which triggers return
Set the xxx IE of the return factor, and set MIE flag in the PSW.
Set HALT/STOP mode
HALT/STOP mode
Watchdog timer
HALT mode : stop counting
STOP mode: clear counter
When returning from STOP mode, wait for oscillation to stabilize.
Return factor interrupt occured
NORMAL mode
Watchdog timer
HALT mode : restart counting
STOP mode: continue counting
Interrupt acceptance cycle

Figure:3.3.1 Transition to/from STANDBY mode

If an interrupt is enabled but interrupt priority level of the interrupt to be used is not equal to or higher than the mask level in PSW before transition to HALT or STOP mode, it is impossible to return to CPU OPERATION mode by maskable interrupt.

After reset is released, wait for 30 ms or longer to change the operation mode to STOP mode.

■ Transition to HALT Modes

The system transfers from RC mode or OSC mode to HALT0 mode, and from PLL mode to HALT2 mode. CPU stops operating, but the oscillators remain operational. There are two ways to return from HALT mode: reset or interrupt. Reset produces normal reset; interrupt, immediate return to the CPU state prior to the transition to HALT mode. The watchdog timer, if enabled, resumes counting.

```
Program 9
              MOV x'04', D0            ; Set HALT mode.
              MOV D0, (CPUM)
              NOP                      ; After written in CPUM, some NOP instructions (three or less) are
              NOP                      ; executed depending on pipeline's condition.
              NOP
```

■ Transition to STOP mode

The system transfers from RC mode or OSC mode to STOP0 mode. The transition from PLL mode to STOP mode cannot be executed. In both cases, oscillation and CPU are both halted. There are two ways to return from STOP mode: reset or interrupt.

On the transition to STOP mode, the counter of watchdog timer is cleared. During recovery, the counting is started and oscillation stabilization wait is generated. After return to CPU OPERATION, the counting is continued.

```
Program 10
              MOV x'08', D0            ; Set STOP mode.
              MOV D0, (CPUM)
              NOP                      ; After written in CPUM, some NOP instructions (three or less) are
              NOP                      ; executed depending on pipeline’s condition.
              NOP
```

Insert three NOP instructions right after the instruction of the transition to HALT and STOP modes.

Transition from PLL mode to STOP mode requires passing through RC mode or OSC mode.

If the generation of the return interrupt factor is not guaranteed after the request for the transition to STANDBY mode (HALT/STOP) by setting of the CPUM register, refer to [Chapter 3 3.3.4 Note for Transition to STANDBY modes].

3.3.4 Note for Transition to STANDBY modes

When the return interrupt factor is not guaranteed to occur after the request for the transition to STANDBY mode by the setting for CPUM register, the following process should be performed for not transiting to STANDBY mode after the occurrence of the return interrupt.

```
#Program for transition to STANDBY mode
        .
        .
        mov   x'08',   D1                  ; Preparation for setting STOP mode
        mov   psw,   D0
        or   x'40',   D0
        mov   D0,   psw                    ; Setting of MIE flag
label_standby:
        mov   D1,   (CPUM)                 ; Setting of STOP mode
        .
        .

#Program for transition to STANDBY mode
        push   D0                          ; Save of register
        push   D1                          ; Save of register
        mov   (x'5',   sp),   D0
        and   x'0F',   D0                  ; Reduction of half byte information
        cmp   ah(label_standby),   D0      ; Comparison of upper address for saving
        bne   next_program
        movw(x'3',   sp),   DW0
        cmp   al(label_standby),   DW0     ; Comparison of lower address for saving
        bne   next_program
        addw   x'02',   DW0                ; Instruction length of mov dn, (CPUM)
        movw   DW0,   (x'3',   sp)         ; Change of address for saving
next_program:
        pop   D1
        pop   D0
```

Chapter 4　Interrupts

4.1 Overview

This LSI speeds up interrupt response with circuitry that automatically loads the branch address to the corresponding interrupt service routine from an interrupt vector table: reset, non-maskable interrupts (NMI), 5 external interrupts, and 24 internal interrupts (peripheral function interrupts).

For interrupts other than reset, the interrupts processing sequence consists of interrupt request, interrupt acceptance determination (only maskable interrupt), interrupt acceptance (hardware processing) and return (RTI instruction). After the interrupt is accepted (hardware processing), the program counter (PC) and processor status word (PSW) and handy addressing data (HA) are saved onto the stack and program is branched to the address specified by the corresponding interrupt vector. Handy addressing data (HA) is saved onto the stack so that it may not be influenced by the interrupt.

And an interrupts handler ends by restoring, using the POP instruction and other means, the contents of any registers used during processing and then executing the return from interrupt (RTI) instruction to return to the point at which execution was interrupted. At least 12 machine cycles for interrupt acceptance, and 11 machine cycles for the return from interrupt.

Each interrupt has a interrupt control register, which controls the interrupts. Interrupt control register consists of the interrupt request flag (IR), interrupt enable flag (IE), and interrupt level specification flag (LV1 to 0).

Interrupt request flag (IR) is set to "1" when an event as an interrupt factor is occurred, and cleared to "0" by the interrupt acceptance. This flag is controlled by hardware, but can be rewritten by software.

Interrupt enable flag (IE) is the flag that enables specified interrupts. There is no interrupt enable flag in non-maskable interrupt (NMI). Once this interrupt request flag is set, it is accepted without any conditions. Interrupts enable flag is set in maskable interrupt. Interrupt enable flag of maskable interrupt is valid when the maskable interrupt enable flag (MIE flag) of PSW is "1".

Maskable interrupts have vector numbers by hardware, but their priority can be changed by user’s program by setting interrupts level specification flags (LV1 to 0). There are three hierarchical interrupt levels. If multiple interrupts have the same priority, the one with the lowest vector number takes priority. Maskable interrupts are accepted when its level is higher than the interrupt mask level (IM1 to 0) of PSW. Non-maskable interrupts are always accepted, regardless of the interrupt mask level.

4.1.1 Functions

Table:4.1.1 Interrupt Functions

Interrupt type		Reset interrupt	Non-maskable interrupt	Maskable interrupt
Vector number		0	1	2 to 30
Table address	IVBM = 0	0x04000	0x04004	0x04008 to 0x04078
	IVBM = 1		0x00104	0x00108 to 0x00178
Starting address		Address specified by vector table		
Interrupt level		-	-	Can be set to level 0 to 2 (by software)
Interrupt factor		External reset pin input	Errors detection, Program interrupt (PI)	External pin input internal peripheral function
Generated operation		Direct input to CPU	Input to CPU from non-maskable interrupt control register (NMICR)	Input interrupt request level set in interrupt level flag (xxxLVn) of maskable interrupt control register (xxxICR) to CPU.
Accept operation		Always accepts	Always accepts	Acceptance is determined by the interrupt control of the PSW interrupt enable flag (MIE), interrupt mask level (IM) and maskable interrupt control register (xxxICR).
Machine cycles until accepted		At least 6 machine cycles + oscillation stabilization wait time	At least 12 machine cycles	At least 12 machine cycles
PSW status after acceptance		All flags are cleared to "0"	The interrupt mask level flag in PSW is cleared to "00"	Values of the interrupt level flag (xxxLVn) are set to the interrupt mask level (masking all interrupt requests with the same or the lower priority).

4.1.2 Block Diagram

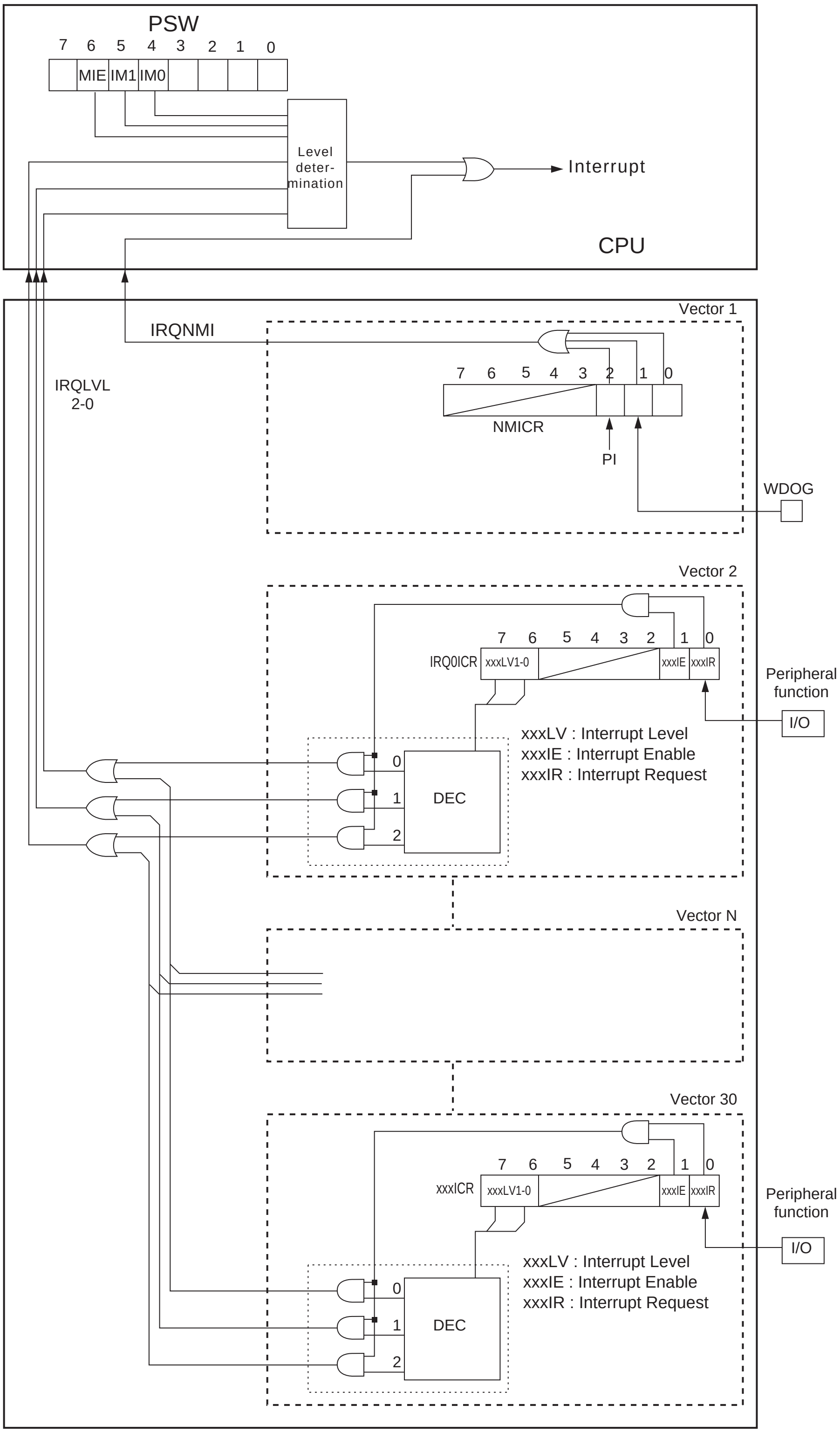


Figure:4.1.1 Interrupt Block Diagram

4.1.3 Operation

■ Interrupt Processing Sequence

For interrupts other than reset, the interrupt processing sequence consists of interrupt request, interrupt acceptance determination (only maskable interrupt), interrupt acceptance (hardware processing), and return (RTI instruction). The program counter (PC) and processor status word (PSW) and hard addressing data (HA) are saved onto the stack, and program is branched to the address specified by the corresponding interrupt vector. An interrupt handler ends by restoring the contents of any registers used during processing and then executing the return from interrupt (RTI) instruction to return to the point at which execution was interrupted.

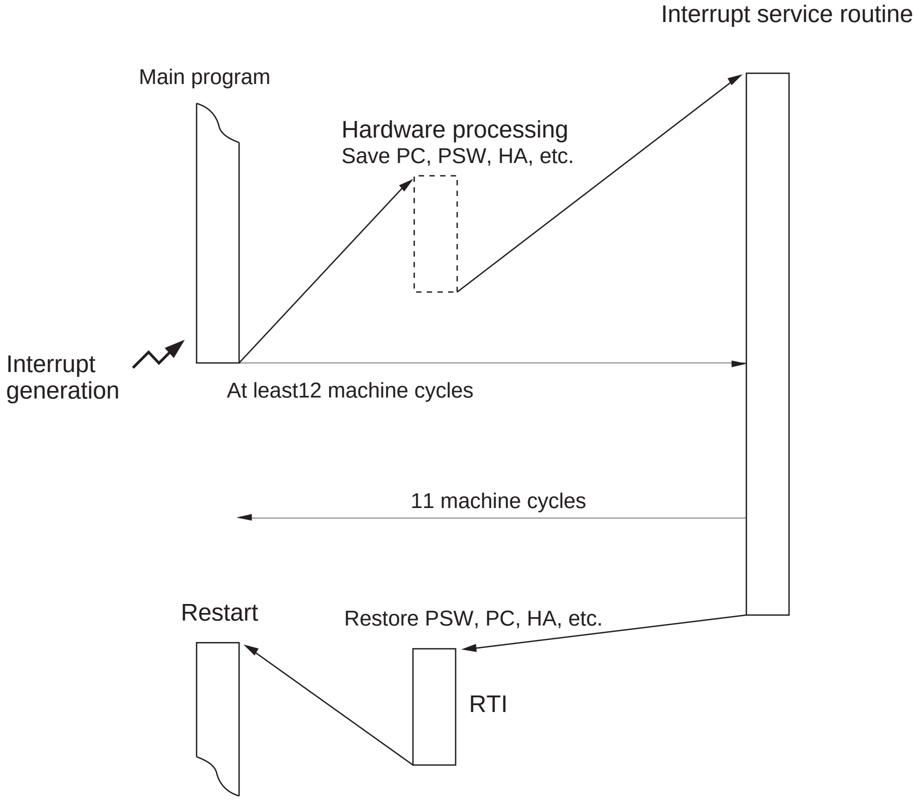


Figure:4.1.2 Interrupt Processing Sequence

The interrupt request flag (IR) of maskable interrupt control register (ICR) is cleared by hardware when an interrupt is accepted. The interrupt request flags of the non-maskable interrupt control register (NMICR) is not cleared by hardware.

■ Interrupt Group and Vector Addresses

Table:4.1.2 shows the list of interrupt vector addresses and interrupt group.

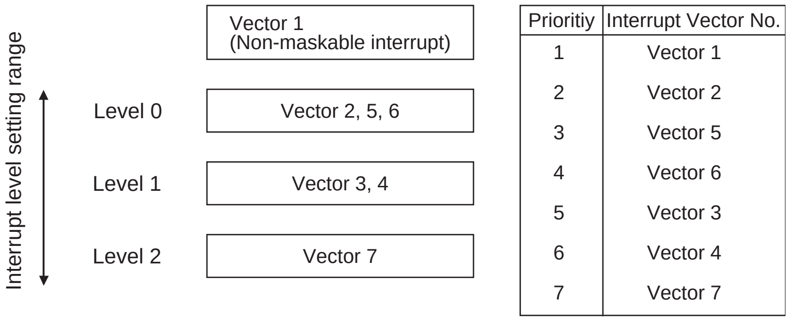
Table:4.1.2 Interrupt Vector Addresses and Interrupt Group

Table remarks √: With function -: Without function

Vector Number	Vector address		Interrupt group (interrupt factor)		Control register (address)		MN101 EFA8	MN101 EFA3	MN101 EFA7	MN101 EFA2
	IVBM=0	IVBM=1								
0	0x04000		Reset	-	-	-	√	√	√	√
1	0x04004	0x00104	Non-maskable interrupt	NMI	NMICR	0x03FE1	√	√	√	√
2	0x04008	0x00108	External Interrupt 0	IRQ0	IRQ0ICR	0x03FE2	√	√	√	√
3	0x0400C	0x0010C	External Interrupt 1	IRQ1	IRQ1ICR	0x03FE3	√	√	√	√
4	0x04010	0x00110	External Interrupt 2	IRQ2	IRQ2ICR	0x03FE4	√	√	√	√
5	0x04014	0x00114	External Interrupt 3	IRQ3	IRQ3ICR	0x03FE5	√	√	√	√
6	0x04018	0x00118	External Interrupt 4 Key interrupt	IRQ4	IRQ4ICR	0x03FE6	√	√	√	√
7	0x0401C	0x0011C	Touch 0 detect interrupt	TS0DTIRQ	TS0DTICR	0x03FE7	√	-	√	-
8	0x04020	0x00120	Touch 0 detect error interrupt	TS0DEIRQ	TS0DEICR	0x03FE8	√	-	√	-
9	0x04024	0x00124	Touch 0 cycle interrupt	TS0CIRQ	TS0CICR	0x03FE9	√	-	√	-
10	0x04028	0x00128	Touch 0 data transmission interrupt	TS0ATIRQ	TS0ATICR	0x03FEA	√	-	√	-
11	0x0402C	0x0012C	Touch 1 detect interrupt	TS1DTIRQ	TS1DTICR	0x03FEB	√	-	-	-
12	0x04030	0x00130	Touch 1 detect error interrupt	TS1DEIRQ	TS1DEICR	0x03FEC	√	-	-	-
13	0x04034	0x00134	Touch 1 round interrupt	TS1CIRQ	TS1CICR	0x03FED	√	-	-	-
14	0x04038	0x00138	Touch 1 data transmission interrupt	TS1ATIRQ	TS1ATICR	0x03FEE	√	-	-	-
15	0x0403C	0x0013C	Timer 0 interrupt	TM0IRQ	TM0ICR	0x03FEF	√	√	√	√
16	0x04040	0x00140	Timer 1 interrupt	TM1IRQ	TM1ICR	0x03FF0	√	√	√	√
17	0x04044	0x00144	Timer 2 interrupt	TM2IRQ	TM2ICR	0x03FF1	√	√	√	√
18	0x04048	0x00148	Timer 3 interrupt	TM3IRQ	TM3ICR	0x03FF2	√	√	√	√
19	0x0404C	0x0014C	Timer 6 interrupt	TM6IRQ	TM6ICR	0x03FF3	√	√	√	√
20	0x04050	0x00150	Time base interrupt	TBIRQ	TBICR	0x03FF4	√	√	√	√
21	0x04054	0x00154	Timer 7 interrupt	TM7IRQ	TM7ICR	0x03FF5	√	√	√	√
22	0x04058	0x00158	Timer 7 compare 2-match interrupt	TM7OC2IRQ	TM7OC2ICR	0x03FF6	√	√	√	√
23	0x0405C	0x0015C	Timer 8 interrupt	TM8IRQ	TM8ICR	0x03FF7	√	√	√	√
24	0x04060	0x00160	Timer 8 compare 2-match interrupt	TM8OC2IRQ	TM8OC2ICR	0x03FF8	√	√	√	√
25	0x04064	0x00164	Timer 9 overflow interrupt	PWMOVIRQ	PWMOVICR	0x03FF9	√	√	√	√
26	0x04068	0x00168	Timer 9 underflow interrupt	PWMUDIRQ	PWMUDICR	0x03FFA	√	√	√	√
27	0x0406C	0x0016C	Serial 0 transmission interrupt	SC0TIRQ	SC0TICR	0x03FFB	√	√	√	√
28	0x04070	0x00170	Serial 0 reception interrupt	SC0RIRQ	SC0RICR	0x03FFC	√	√	√	√
29	0x04074	0x00174	A/D conversion interrupt	ADIRQ	ADICR	0x03FFD	√	√	√	√
30	0x04078	0x00178	Peripheral function group interrupt	PERIIRQ	PERIILR	0x03FFE	√	√	√	√

■ Interrupt Level and Priority

In this LSI, vector numbers and interrupt control registers (except reset interrupt) are allocated to each interrupt. The interrupt level (except reset interrupt, non-maskable interrupt) can be set by software, per interrupt group. There are three hierarchical interrupt levels. If multiple interrupts have the same priority, the one with the lowest vector number takes priority. For example, if both vector 3 (level 1) and vector 4 (level 1) request interrupt simultaneously, vector 3 will be accepted.



Prioritiy	Interrupt Vector No.
1	Vector 1
2	Vector 2
3	Vector 5
4	Vector 6
5	Vector 3
6	Vector 4
7	Vector 7

Figure:4.1.3 Interrupt Priority Outline

Table:4.1.3 Interrupt Mask Level and Interrupt Acceptance

	Interrupt mask level		Priority	Acceptable interrupt level
	IM1	IM0		
Mask level 0	0	0	Highest	Non-maskable interrupt (NMI) only
Mask level 1	0	1	:	NMI, level 0
Mask level 2	1	0	:	NMI, level 0 to 1
Mask level 3	1	1	Lowest	NMI, level 0 to 2

■ Determination of Maskable Interrupt Acceptance

The following is the procedure from interrupt request input to acceptance.

1. The interrupt request flag (xxxIR) in the corresponding external interrupt control register (IRQnICR) and internal interrupt control register (xxxICR) are set to "1".
2. When the interrupt enable flag (xxxIE) corresponding to the internal request flag is "1", the information of the level set in the interrupt level flag (xxxLV1 to 0) is output to the CPU as an interrupt request signal.
3. The interrupt request is accepted if the output interrupt request signal has a higher priority than the level set in the internal mask level of the processor status word (PSW) and the interrupt enable flag of PSW (MIE) is "1" (enabled).
4. After acceptance of an interrupt, the interrupt request flag (xxxIR) is cleared by hardware. However, the interrupt enable flag (xxxIE) is not cleared.

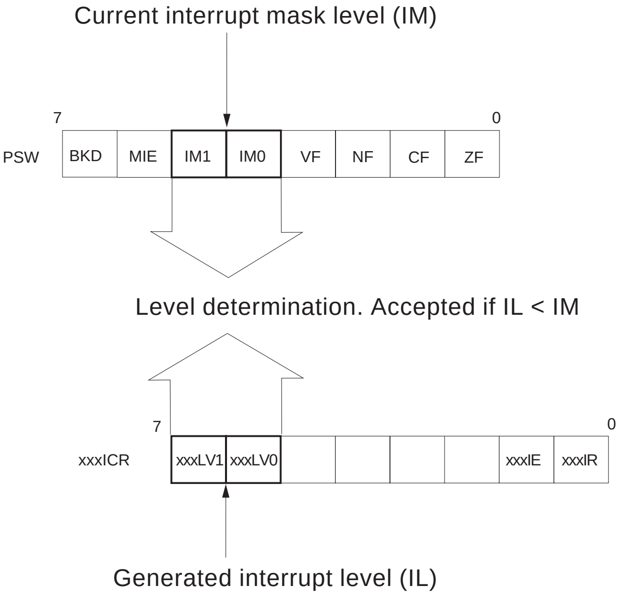

Current interrupt mask level (IM)
7
0
PSW
BKD
MIE
IM1
IM0
VF
NF
CF
ZF
Level determination. Accepted if IL < IM
7
0
xxxICR
xxxLV1
xxxLV0
xxxIE
xxxIR
Generated interrupt level (IL)

Figure:4.1.4 Determination of Interrupt Acceptance

After acceptance of interrupt, the interrupt enable flag (xxxIE) is not cleared.

After acceptance of interrupt, the interrupt with the same factor will be ignored until the interrupt request flag (xxxIR) is cleared by hardware.

■ Maskable interrupt Enable (MIE) and Interrupt Mask level (IM1-0) in PSW

MIE = "0" and maskable interrupts are disabled when:

- MIE in the PSW is reset to "0" by a program
- BE instruction is executed. (BKD is reset and MIE is reset)
- Reset is detected.

MIE = "1" and maskable interrupts are enabled when:

- MIE in the PSW is reset to "1" by a program
- BE instruction is executed. (BKD is set and MIE is set)

The interrupt mask level (IM=IM1 to IM0) in processor status word (PSW) changes when:

- The program alters it directly.
- Reset is detected. (IM="00")
- Maskable interrupt is accepted (interrupt level becomes the interrupt mask level. Execution of the RTI instruction at the end of an interrupt service routine restores the processor status word (PSW) and thus the previous interrupt mask level.).
- Non-Maskable interrupt is accepted (IM="00").

When accepting interrupts, MIE of PSW is not cleared to "0".

When a non-maskable interrupt and a maskable interrupt are generated at the same time, the non-maskable interrupt has a priority to the maskable one.

As for BE instruction and BD instruction, refer to [Chapter 17 17.2 Instruction Set].

■ Interrupt Acceptance Operation (Hardware Processing)

When accepting an interrupt, this LSI hardware saves the return address from program counter, and processor status word (PSW) to the stack and branches program to the interrupt handler using the starting address in vector table. The following is hardware processing sequence invoked by interrupt acceptance.

1. Stack pointer (SP) is updated.

   SP-6 → SP

2. The contents of the program counter (PC)-i.e., the return address- are saved to the stack.

   PC bits 7 to 0 → Address (SP + 1)

   PC bits 15 to 8 → Address (SP + 2)

   PC bits 19 to 16, and H → Address (SP + 3)

3. The contents of handy addressing data (HA) are saved to the stack.

   Lower half of HA → Address (SP + 4)

   Upper half of HA → Address (SP + 5)

4. The contents of PSW are saved to the stack.

   PSW → Address (SP)

5. Interrupt level (xxxLVn) for the interrupt is copied to interrupt mask IMn in PSW.

   Interrupt level (xxxLVn) → IMn

6. BKD flag of PSW is reset (When accepting interrupts, bank register always address the first 64 KB.)

7. Hardware branches program to the address in vector table.

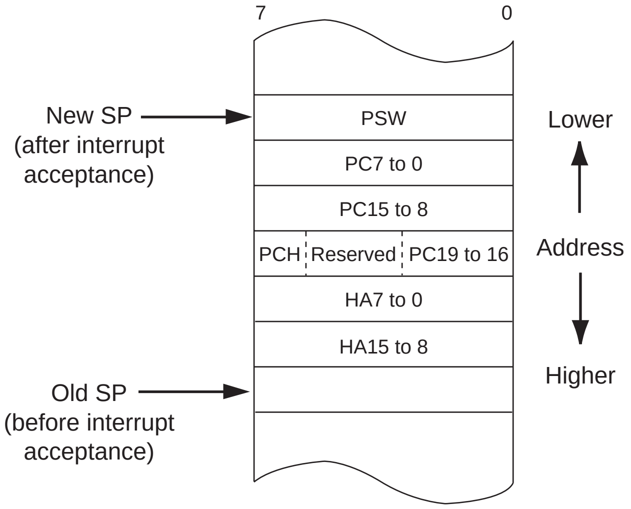


Figure:4.1.5 Stack Operation during Interrupt Acceptance

■ Interrupt Return Operation (RTI Instruction)

An interrupt handler ends by restoring the contents of any registers saved to the stack during processing by POP instruction and other means, and RTI instruction restores the program to the point at execution was interrupted.

The following is processing sequence invoked by RTI instruction.

1. The contents of PSW are restored from stack. (SP)
2. The contents of program counter (PC) -i.e., the return address- are restored from stack. (SP + 1 to SP + 3)
3. The contents of handy addressing data (HA) are restored from stack. (SP + 4, SP+ 5)
4. Stack pointer is updated. SP+6 → SP
5. Execution branches program to the address in program counter.

The handy addressing data is an internal register used by the handy addressing function. The hardware saves its contents to the stack to prevent the interrupt from interfering with operation of the function.

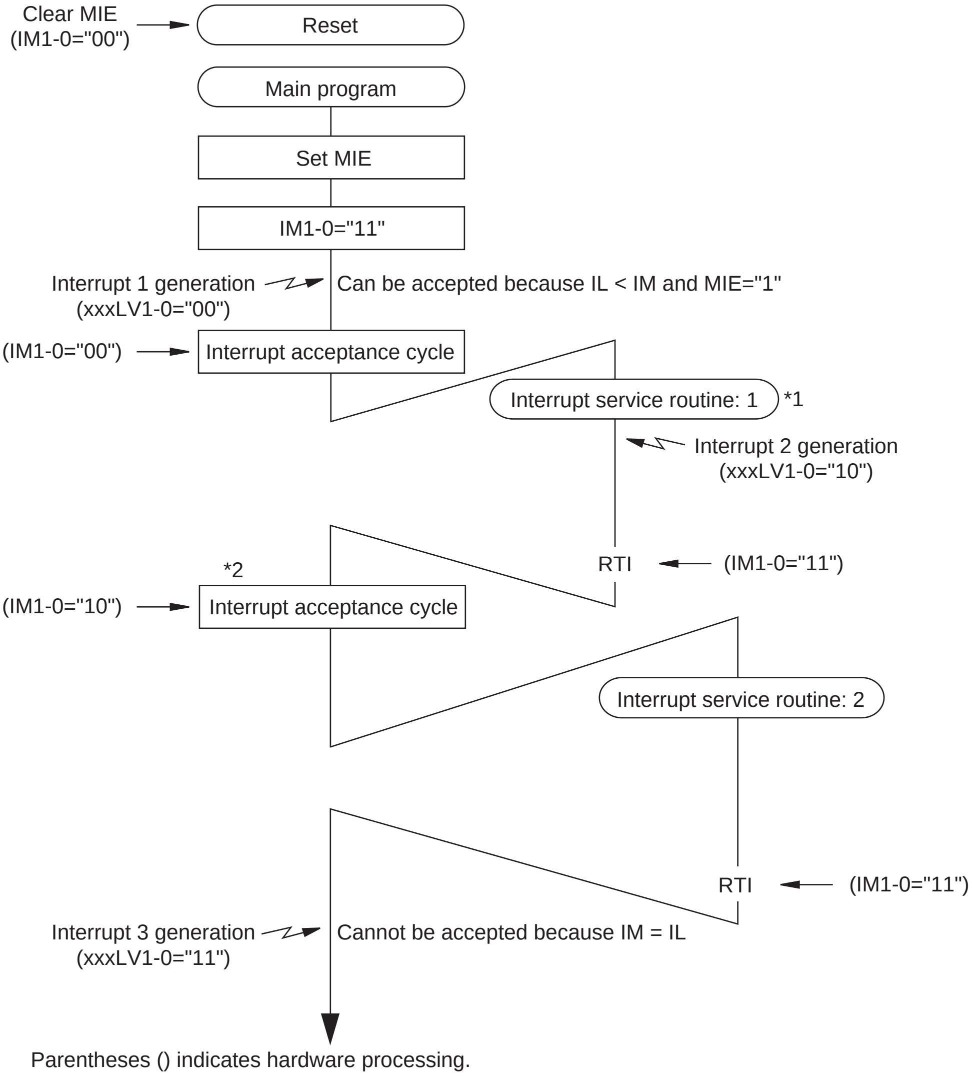
Registers including data registers and address registers are not saved. If needed, save these registers to stack using PUSH instructions.

In Figure:4.1.5, bp6 to bp4 of stack address where program counter (PC [bit19 to bit16, and bitH]) are saved, are reserved. Do not change it by program.

■ Maskable Interrupt

The following is the processing sequence when the lower priority level interrupt occurs while processing the higher priority level interrupt.

(Interrupt 1: xxxLV1-0="00", Interrupt 2: xxxLV1-0="10", Interrupt 2: xxxLV1-0="11")



Parentheses () indicates hardware processing.

*1: If IL < IM, an interrupt generated in the interrupt processing program 1 is accepted as multiple interrupt.
If IL > IM, however, the interrupt is not accepted.

*2: If the interrupt 2 generated in the interrupt processing program 1 is not accepeted when IL ≥ IM,
the interrupt 2 will be accepted after completion of the interrupt processing program 1.

Figure:4.1.6 Processing Sequence for Maskable Interrupts

■ Multiple Interrupt of Maskable Interrupt

This LSI automatically disables acceptance of subsequent interrupts with the same or lower priority level when accepting an interrupt. The LSI copies interrupt level (xxxLVn) for the interrupt to interrupt mask (IM) in PSW when the hardware accepts an interrupt. As a result, subsequent interrupts with the same or lower priority levels are automatically masked. Only interrupts with higher priority levels are accepted. The net result is that interrupts are normally processed in decreasing order of priority. It is, however, possible to alter this arrangement.

1. To disable interrupt nesting
   - Reset the MIE bit in PSW to "0".
   - Raise the priority level of interrupt mask (IM) in PSW.

   Execute either one of the operation above.
2. To enable interrupts with lower priority than the currently accepted interrupt
   - Lower the priority level or interrupt mask (IM) in PSW.

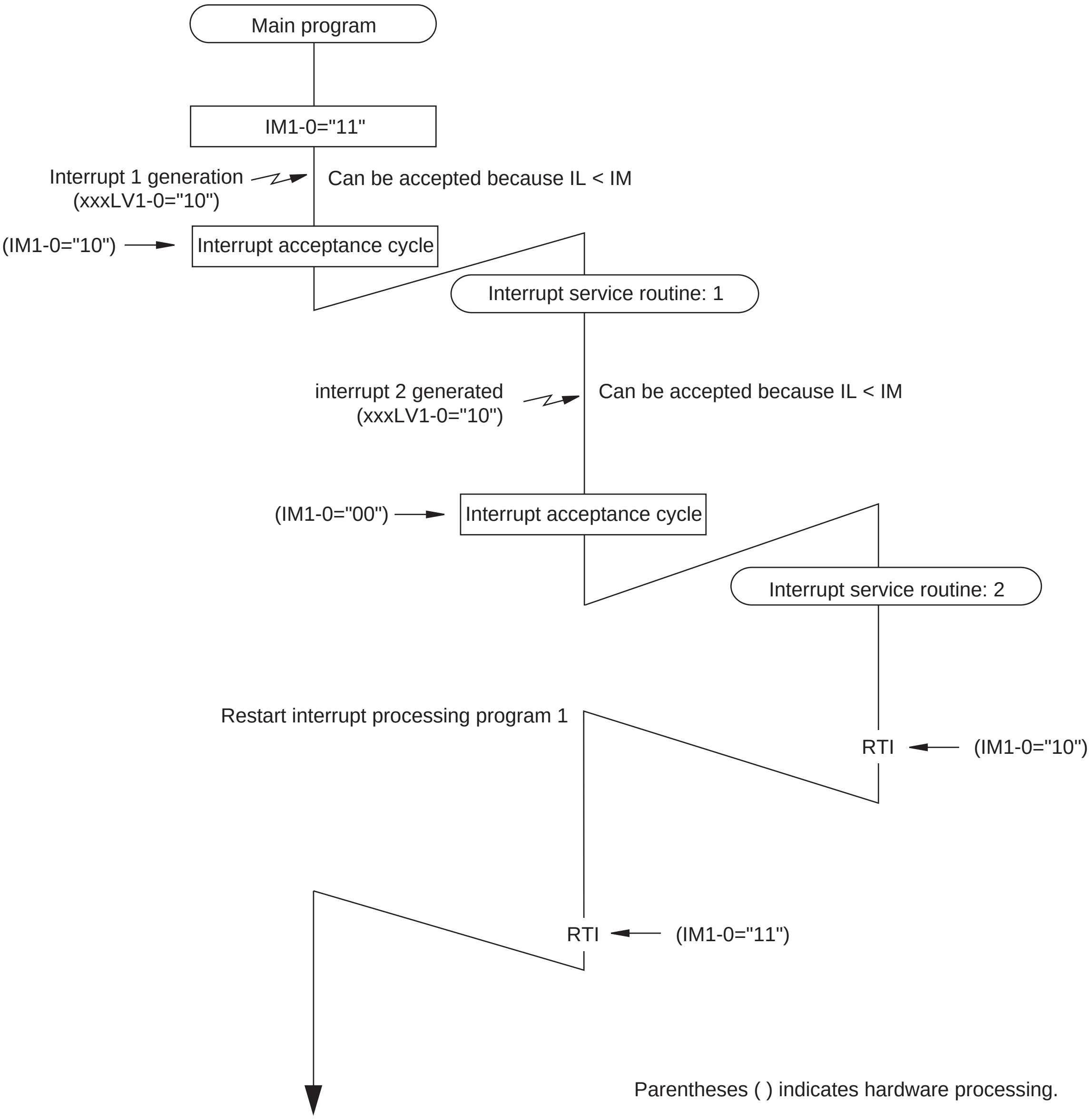
Multiple interrupts are enabled only for interrupts with levels higher than PSW interrupt mask level (IM).

It is possible to forcibly rewrite IM to accept an interrupt with a priority lower than the interrupt being processed. In this case, be careful of stack overflow caused by multiple interrupt.

Do not operate maskable interrupt control register (xxxICR) when multiple interrupts are enabled. If the operation is necessary, clear the MIE flag of PSW once to disable the interrupts.

Figure:4.1.7 shows the processing sequence of multiple interrupt.

(Multiple interrupt 1: xxxLV1 to 0 = "10", Multiple interrupt 2: xxxLV1 to 0 = "00")


Main program
IM1-0="11"
Interrupt 1 generation
(xxxLV1-0="10")
Can be accepted because IL < IM
(IM1-0="10")
Interrupt acceptance cycle
Interrupt service routine: 1
interrupt 2 generated
(xxxLV1-0="10")
Can be accepted because IL < IM
(IM1-0="00")
Interrupt acceptance cycle
Interrupt service routine: 2
Restart interrupt processing program 1
RTI
(IM1-0="10")
RTI
(IM1-0="11")

Parentheses ( ) indicates hardware processing.

Figure:4.1.7 Processing Sequence for Multiple Maskable Interrupts

■ Multiple Non-maskable Interrupt (NMI)

On the acceptance of NMI, when other NMI factor is generated, this interrupt is processed immediately.
Also, the same NMI factor is not accepted if it is generated before the NMI flag is cleared by the software.
(Unless the NMI flag is cleared by software, otherwise the next same NMI interrupt is not accepted and becomes invalid.)

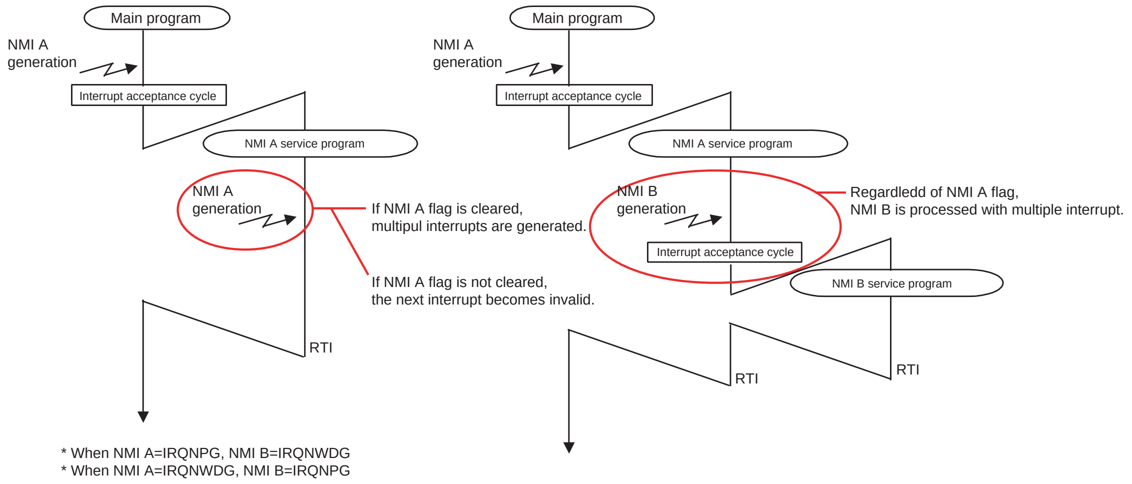

Main program
NMI A generation
Interrupt acceptance cycle
NMI A service program
NMI A generation
If NMI A flag is cleared, multipul interrupts are generated.
If NMI A flag is not cleared, the next interrupt becomes invalid.
RTI
Main program
NMI A generation
Interrupt acceptance cycle
NMI A service program
NMI B generation
Interrupt acceptance cycle
Regardledd of NMI A flag, NMI B is processed with multiple interrupt.
NMI B service program
RTI
RTI
* When NMI A=IRQNPG, NMI B=IRQNWDG
* When NMI A=IRQNWDG, NMI B=IRQNPG

Figure:4.1.8 Processing Sequence for Multiple Interrupt of Non-maskable

4.1.4 Maskable Interrupt Control Register Setup

■ Interrupt Request Flag (IR) Setup by Software

The interrupt request flag is operated by hardware. The flag is set to "1" when an interrupt factor is generated, and cleared to "0" when the interrupt is accepted. To rewrite the interrupt request flag by software, set the IRWE flag of the MEMCTR register.

■ Maskable Interrupt Control Register Setup Procedure

A setup procedure of the maskable interrupt control register including change of the interrupt request flag by software is shown below.

Setup Procedure	Description
(1) Disable all maskable interrupts. PSW bp6: MIE =0	(1) Clear the MIE flag of PSW to disable all maskable interrupts. This is necessary, especially when the interrupt control register is changed.
(2) Select the interrupt factor.	(2) Select the interrupt factor such as interrupt edge or timer interrupt cycle.
(3) Enable writing in the interrupt request flag. MEMCTR(0x03F01) bp2: IRWE =1	(3) Set the IRWE flag of memory control register (MEMCTR) to enable writing of the interrupt request flag. This is necessary only when the interrupt request flag is changed by the software.
(4) Rewrite the interrupt request flag. xxxICR bp0: xxxIR	(4) Rewrite the interrupt request flag (xxxIR) of interrupt control register (xxxICR). (The interrupt request flag has already been set. In such case, clear it in this way.)
(5) Disable writing of the interrupt request flag. MEMCTR(0x03F01) bp2: IRWE =0	(5) Clear the IRWE flag to disable writing of the interrupt request flag by the software.
(6) Set the interrupt level. xxxICR bp7 to 6: xxxLV1 to 0 PSW bp5 to 4: IM1 to 0	(6) Set the interrupt level by the xxxLV1 to 0 flag of interrupt control register (xxxICR). If it is necessary to change the interrupt mask level of PSW, set the IM1 to 0 flags of PSW.
(7) Enable the interrupt. xxxICR bp1: xxxIE =1	(7) Set the xxxIE flag of interrupt control register (xxxICR) to enable the interrupt.
(8) Enable all maskable interrupts. PSW bp6: MIE =1	(8) Enable all maskable interrupts.

! The interrupt request flag of the interrupt control register is set by the interrupt generation, the edge switching and others, regardless of the xxxIE flag. Clear the flag in accordance with the setup procedures (3) to (5).

! Be sure to set the IRWE flag of memory control register (MEMCTR) to “0” except in writing IR by software. If interrupt control register (xxxICR) is written by software while the IRWE flag is set to “1”, the interrupt request flag set to “1” by interrupt factor may be cleared.
For example, if xxxICR register is set using bit operation instruction, the value in the register is read, modified and written back to the register.
When an interrupt is generated between the reading and writing the register, the IR flag may be cleared to “0” accidentally and the interrupt processing is not executed.
If the IRWE flag is set to “0”, the IR flag is not modified when writing to xxxICR register.

! Disable all maskable interrupt for the maskable interrupt enable flag (MIE) of processor status word (PSW) (Set the MIE flag to "0") before writing to interrupt control register (xxxICR). There is no guarantee of proper operation when writing to xxxICR register with the MIE flag set to "1".

4.2 Control Registers

4.2.1 Registers List

Table:4.2.1 Interrupt Control Registers

Table remarks √: With function -: Without function

Register	Address	R/W	Function	Page	MN101 EFA8	MN101 EFA3	MN101 EFA7	MN101 EFA2
NMICR	0x03FE1	R/W	Non-maskable interrupt control register	IV-20	√	√	√	√
IRQ0ICR	0x03FE2	R/W	External interrupt 0 control register	IV-21	√	√	√	√
IRQ1ICR	0x03FE3	R/W	External interrupt 1 control register	IV-21	√	√	√	√
IRQ2ICR	0x03FE4	R/W	External interrupt 2 control register	IV-21	√	√	√	√
IRQ3ICR	0x03FE5	R/W	External interrupt 3 control register	IV-21	√	√	√	√
IRQ4ICR	0x03FE6	R/W	External interrupt 4 control register	IV-21	√	√	√	√
TS0DTICR	0x03FE7	R/W	Touch 0 detect interrupt control register	IV-31	√	-	√	-
TS0DEICR	0x03FE8	R/W	Touch 0 detect error interrupt control register	IV-32	√	-	√	-
TS0CICR	0x03FE9	R/W	Touch 0 round interrupt control register	IV-33	√	-	√	-
TS0ATICR	0x03FEA	R/W	Touch 0 data transmission interrupt control register	IV-34	√	-	√	-
TS1DTICR	0x03FEB	R/W	Touch 1 detect interrupt control register	IV-31	√	-	-	-
TS1DEICR	0x03FEC	R/W	Touch 1 detect error interrupt control register	IV-32	√	-	-	-
TS1CICR	0x03FED	R/W	Touch 1 round interrupt control register	IV-33	√	-	-	-
TS1ATICR	0x03FEE	R/W	Touch 1 data transmission interrupt control register	IV-34	√	-	-	-
TM0ICR	0x03FEF	R/W	Timer 0 interrupt control register	IV-22	√	√	√	√
TM1ICR	0x03FF0	R/W	Timer 1 interrupt control register	IV-22	√	√	√	√
TM2ICR	0x03FF1	R/W	Timer 2 interrupt control register	IV-22	√	√	√	√
TM3ICR	0x03FF2	R/W	Timer 3 interrupt control register	IV-22	√	√	√	√
TM6ICR	0x03FF3	R/W	Timer 6 interrupt control register	IV-22	√	√	√	√
TBICR	0x03FF4	R/W	Time base interrupt control register	IV-23	√	√	√	√
TM7ICR	0x03FF5	R/W	Timer 7 interrupt control register	IV-24	√	√	√	√
TM7OC2ICR	0x03FF6	R/W	Timer 7 compare 2-match interrupt control register	IV-25	√	√	√	√
TM8ICR	0x03FF7	R/W	Timer 8 interrupt control register	IV-24	√	√	√	√
TM8OC2ICR	0x03FF8	R/W	Timer 8 compare 2-match interrupt control register	IV-25	√	√	√	√
PWMOVICR	0x03FF9	R/W	Timer 9 overflow interrupt control register	IV-26	√	√	√	√
PWMUDICR	0x03FFA	R/W	Timer 9 underflow interrupt control register	IV-27	√	√	√	√
SC0TICR	0x03FFB	R/W	Serial 0 transmission interrupt control register	IV-28	√	√	√	√
SC0RICR	0x03FFC	R/W	Serial 0 UART reception interrupt control register	IV-29	√	√	√	√
ADICR	0x03FFD	R/W	A/D conversion interrupt control register	IV-30	√	√	√	√

Table remarks √: With function -: Without function

Register	Address	R/W	Function	Page	MN101 EFA8	MN101 EFA3	MN101 EFA7	MN101 EFA2
PERIILR	0x03FFE	R/W	Peripheral function group interrupt level setting register	IV-35	√	√	√	√
IRQEXPEN	0x03F4E	R/W	Peripheral function group input enable register	IV-36	√	√	√	√
IRQEXPDT	0x03F4F	R/W	Peripheral function group interrupt factor retention register	IV-37	√	√	√	√

When the interrupt level flag (xxxLVn) is set to "level 3", its vector is disabled, regardless of the interrupt enable flag and the interrupt request flag.

Disable all maskable interrupt for the maskable interrupt enable flag (MIE) of processor status word (PSW) (Set the MIE flag to "0") before writing to interrupt control register (xxxICR). There is no guarantee of proper operation when writing to xxxICR register with the MIE flag set to "1".

4.2.2 Interrupt Control Registers

The interrupt control registers include non-maskable interrupt control register (NMICTR) and maskable interrupt control register. Maskable interrupt control register consists of external interrupt control register (IRQnICR) and internal interrupt control register (xxxICR).

■ Non-maskable Interrupt Control Register (NMICR: 0x03FE1)

The non-maskable interrupt control register (NMICTR) is stored in the non maskable interrupt request. When the non-maskable interrupt request is generated, the interrupt is accepted regardless of the interrupt mask level (IMn) of PSW. The hardware then branches program to the address stored at location 0x04004 in the interrupt vector table. The watchdog timer overflow interrupt request flag (IRQNWDG) is set to "1" when the watchdog timer overflows.

The program interrupt request flag (IRQNPG) is set to "1" when the undefined instruction is executed.

The peripheral function non maskable interrupt request flag (IRQNPRI) is set to be "1" when a write error interrupt is generated in the register protect function.

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	-	IRQNPG	IRQNWDG	IRQNPRI
At reset	-	-	-	-	-	0	0	0
Access	-	-	-	-	-	R/W	R/W	R/W

bp	Flag	Description
7-3	-	-
2	IRQNPG	Program interrupt request flag 0: No interrupt request has been generated. 1: Interrupt request has been generated.
1	IRQNWDG	Watchdog interrupt request flag 0: No interrupt request has been generated. 1: Interrupt request has been generated.
0	IRQNPRI	Peripheral function non-maskable interrupt request flag (Interrupt by register protect function) 0: No interrupt request has been generated. 1: Interrupt request has been generated.

When the undefined instruction is going to be executed, this LSI generates the non-maskable interrupt at the same time of the setting of the IRQNPG flag.
When the setting of the IRQNPG flag is confirmed by the non-maskable interrupt process program, the software reset is recommended by outputting "0" to the reset pin P27.

■ External Interrupt 0 to 4 Control Registers (IRQ0ICR: 0x03FE2, IRQ1ICR: 0x03FE3, IRQ2ICR: 0x03FE4, IRQ3ICR: 0x03FE5, IRQ4ICR: 0x03FE6)

External interrupt 0 to 4 control registers (IRQ0ICR to IRQ4ICR) are used to control interrupt levels of external interrupts 0 to 4, active edge, interrupt enable and interrupt request. Operate interrupt control register when the maskable interrupt enable flag (MIE) of PSW is "0". Refer to [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup] for the setting procedure of the flags.

bp	7	6	5	4	3	2	1	0
Flag	IRQnLV1	IRQnLV0	REDGn	-	Reserved	-	IRQnIE	IRQnIR
At reset	0	0	0	-	0	-	0	0
Access	R/W	R/W	R/W	-	R/W	-	R/W	R/W

bp	Flag	Description
7-6	IRQnLV1-0	Interrupt level flag This flag sets the interrupt level 0 to 3 for interrupt requests.
5	REDGn	Interrupt valid edge flag (in STANDBY mode, level interrupt selection) 0: Falling edge (Low-level) 1: Rising edge (High-level)
4	-	-
3	Reserved	Always set to “0”.
2	-	-
1	IRQnIE	Interrupt enable flag 0: Disabled 1: Enabled
0	IRQnIR	Interrupt request flag 0: No interrupt request has been generated. 1: Interrupt request has been generated.

For the interrupt generation factors of the external interrupt 0 to 4/Key interrupt in STANDBY mode, refer to [Chapter 4 4.3.8 External Interrupt Operation in STANDBY Mode].

■ Timer 0 to 3, 6 Interrupt Control Registers (TM0ICR: 0x03FEF, TM1ICR: 0x03FF0, TM2ICR: 0x03FF1, TM3ICR: 0x03FF2, TM6ICR: 0x03FF3)

Timer 0 to 3, 6 interrupt control registers (TM0ICR to TM3ICR, TM6ICR) control interrupt levels of Timer 0 to 3, 6 interrupts, interrupt enable and interrupt request. Operate interrupt control register when the maskable interrupt enable flag (MIE) of PSW is "0". Refer to [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup] for the setting procedure of the flags.

bp	7	6	5	4	3	2	1	0
Flag	TMnLV1	TMnLV0	-	-	Reserved	-	TMnIE	TMnIR
At reset	0	0	-	-	0	-	0	0
Access	R/W	R/W	-	-	R/W	-	R/W	R/W

bp	Flag	Description
7-6	TMnLV1-0	Interrupt level flag This flag sets the interrupt level 0 to 3 for interrupt requests.
5-4	-	-
3	Reserved	Always set to “0”.
2	-	-
1	TMnIE	Interrupt enable flag 0: Disabled 1: Enabled
0	TMnIR	Interrupt request flag 0: No interrupt request has been generated. 1: Interrupt request has been generated.

■ Time Base Interrupt Control Register (TBICR: 0x03FF4)

Time base interrupt control register (TBICR) controls interrupt level of time base interrupt, interrupt enable and interrupt request. Operate interrupt control register when the maskable interrupt enable flag (MIE) of PSW is "0". Refer to [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup] for the setting procedure of the flags.

bp	7	6	5	4	3	2	1	0
Flag	TBLV1	TBLV0	-	-	Reserved	-	TBIE	TBIR
At reset	0	0	-	-	0	-	0	0
Access	R/W	R/W	-	-	R/W	-	R/W	R/W

bp	Flag	Description
7-6	TBLV1-0	Interrupt level flag This flag sets the interrupt level 0 to 3 for interrupt requests.
5-4	-	-
3	Reserved	Always set to “0”.
2	-	-
1	TBIE	Interrupt enable flag 0: Disabled 1: Enabled
0	TBIR	Interrupt request flag 0: No interrupt request has been generated. 1: Interrupt request has been generated.

■ Timer 7, 8 Interrupt Control Register (TM7ICR: 0x03FF5, TM8ICR: 0x03FF7)

Timer 7, 8 interrupt control register (TM7ICR,TM8ICR) controls interrupt level of timer 7, 8 interrupt, interrupt enable and interrupt request. Operate interrupt control register when the maskable interrupt enable flag (MIE) of PSW is "0". Refer to [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup] for the setting procedure of the flags.

bp	7	6	5	4	3	2	1	0
Flag	TMnLV1	TMnLV0	-	-	Reserved	-	TMnIE	TMnIR
At reset	0	0	-	-	0	-	0	0
Access	R/W	R/W	-	-	R/W	-	R/W	R/W

bp	Flag	Description
7-6	TMnLV1-0	Interrupt level flag This flag sets the interrupt level 0 to 3 for interrupt requests.
5-4	-	-
3	Reserved	Always set to “0”.
2	-	-
1	TMnIE	Interrupt enable flag 0: Disabled 1: Enabled
0	TMnIR	Interrupt request flag 0: No interrupt request has been generated. 1: Interrupt request has been generated.

■ Timer 7, 8 Compare 2-Match Interrupt Control Register (TM7OC2ICR: 0x03FF6, TM8OC2ICR: 0x03FF8)

Timer 7, 8 compare 2-match interrupt control register (TM7OC2ICR,TM8OC2ICR) controls interrupt level of timer 7, 8 compare 2-match interrupt, interrupt enable and interrupt request. Operate interrupt control register when the maskable interrupt enable flag (MIE) of PSW is "0". Refer to [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup] for the setting procedure of the flags.

bp	7	6	5	4	3	2	1	0
Flag	TMnOC2 LV1	TMnOC2 LV0	-	-	Reserved	-	TMnOC2 IE	TMnOC2 IR
At reset	0	0	-	-	0	-	0	0
Access	R/W	R/W	-	-	R/W	-	R/W	R/W

bp	Flag	Description
7-6	TMnOC2LV1-0	Interrupt level flag This flag sets the interrupt level 0 to 3 for interrupt requests.
5-4	-	-
3	Reserved	Always set to “0”.
2	-	-
1	TMnOC2IE	Interrupt enable flag 0: Disabled 1: Enabled
0	TMnOC2IR	Interrupt request flag 0: No interrupt request has been generated. 1: Interrupt request has been generated.

Do not word access to this register.

■ Timer 9 Overflow Interrupt Control Register (PWMOVICR: 0x03FF9)

Timer 9 overflow interrupt control register (PWMOVICR) controls interrupt level of timer 9 overflow interrupt, interrupt enable and interrupt request. Operate interrupt control register when the maskable interrupt enable flag (MIE) of PSW is "0". Refer to [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup] for the setting procedure of the flags.

bp	7	6	5	4	3	2	1	0
Flag	PWMOV LV1	PWMOV LV0	-	-	Reserved	-	PWMOV IE	PWMOV IR
At reset	0	0	-	-	0	-	0	0
Access	R/W	R/W	-	-	R/W	-	R/W	R/W

bp	Flag	Description
7-6	PWMOVLV1-0	Interrupt level flag This flag sets the interrupt level 0 to 3 for interrupt requests.
5-4	-	-
3	Reserved	Always set to “0”.
2	-	-
1	PWMOVIE	Interrupt enable flag 0: Disabled 1: Enabled
0	PWMOVIR	Interrupt request flag 0: No interrupt request has been generated. 1: Interrupt request has been generated.

■ Timer 9 Underflow Interrupt Control Register (PWMUDICR: 0x03FFA)

Timer 9 underflow interrupt control register (PWMUDICR) controls interrupt level of timer 9 underflow interrupt, interrupt enable and interrupt request. Operate interrupt control register when the maskable interrupt enable flag (MIE) of PSW is "0". Refer to [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup] for the setting procedure of the flags.

bp	7	6	5	4	3	2	1	0
Flag	PWMUD LV1	PWMUD LV0	-	-	Reserved	-	PWMUD IE	PWMUD IR
At reset	0	0	-	-	0	-	0	0
Access	R/W	R/W	-	-	R/W	-	R/W	R/W

bp	Flag	Description
7-6	PWMUDLV1-0	Interrupt level flag This flag sets the interrupt level 0 to 3 for interrupt requests.
5-4	-	-
3	Reserved	Always set to “0”.
2	-	-
1	PWMUDIE	Interrupt enable flag 0: Disabled 1: Enabled
0	PWMUDIR	Interrupt request flag 0: No interrupt request has been generated. 1: Interrupt request has been generated.

■ Serial 0 Transmission Interrupt Control Register (SC0TICR: 0x03FFB)

Serial 0 transmission interrupt control registers (SC0TICR) control interrupt levels of Serial 0 transmission interrupts, interrupt enable and interrupt request. Operate interrupt control register when the maskable interrupt enable flag (MIE) of PSW is "0". Refer to [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup] for the setting procedure of the flags.

bp	7	6	5	4	3	2	1	0
Flag	SC0TLV1	SC0TLV0	-	-	Reserved	-	SC0TIE	SC0TIR
At reset	0	0	-	-	0	-	0	0
Access	R/W	R/W	-	-	R/W	-	R/W	R/W

bp	Flag	Description
7-6	SC0TLV1-0	Interrupt level flag This flag sets the interrupt level 0 to 3 for interrupt requests.
5-4	-	-
3	Reserved	Always set to “0”.
2	-	-
1	SC0TIE	Interrupt enable flag 0: Disabled 1: Enabled
0	SC0TIR	Interrupt request flag 0: No interrupt request has been generated. 1: Interrupt request has been generated.

■ Serial 0 UART Reception Interrupt Control Register (SC0RICR: 0x03FFC)

Serial 0 UART reception interrupt control registers (SC0RICR) control interrupt level of serial 0 UART reception interrupt, interrupt enable and interrupt request. Operate interrupt control register when the maskable interrupt enable flag (MIE) of PSW is "0". Refer to [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup] for the setting procedure of the flags.

bp	7	6	5	4	3	2	1	0
Flag	SC0RLV1	SC0RLV0	-	-	Reserved	-	SC0RIE	SC0RIR
At reset	0	0	-	-	0	-	0	0
Access	R/W	R/W	-	-	R/W	-	R/W	R/W

bp	Flag	Description
7-6	SC0RLV1-0	Interrupt level flag This flag sets the interrupt level 0 to 3 for interrupt requests.
5-4	-	-
3	Reserved	Always set to “0”.
2	-	-
1	SC0RIE	Interrupt enable flag 0: Disabled 1: Enabled
0	SC0RIR	Interrupt request flag 0: No interrupt request has been generated. 1: Interrupt request has been generated.

■ A/D Conversion Interrupt Control Register (ADICR: 0x03FFD)

A/D conversion interrupt control register (ADICR) controls interrupt level of A/D conversion interrupt, interrupt enable and interrupt request. Operate interrupt control register when the maskable interrupt enable flag (MIE) of PSW is "0". Refer to [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup] for the setting procedure of the flags.

bp	7	6	5	4	3	2	1	0
Flag	ADLV1	ADLV0	-	-	Reserved	-	ADIE	ADIR
At reset	0	0	-	-	0	-	0	0
Access	R/W	R/W	-	-	R/W	-	R/W	R/W

bp	Flag	Description
7-6	ADLV1-0	Interrupt level flag This flag sets the interrupt level 0 to 3 for interrupt requests.
5-4	-	-
3	Reserved	Always set to “0”.
2	-	-
1	ADIE	Interrupt enable flag 0: Disabled 1: Enabled
0	ADIR	Interrupt request flag 0: No interrupt request has been generated. 1: Interrupt request has been generated.

■ Touch 0, 1 Detect Interrupt Control Register (TS0DTICR: 0x03FE7, TS1DTICR: 0x03FEB)

Touch 0, 1 detect interrupt control register (TS0DTICR, TS1DTICR) controls interrupt level of touch 0, 1 detect interrupt, interrupt enable and interrupt request. Operate interrupt control register when the maskable interrupt enable flag (MIE) of PSW is "0". Refer to [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup] for the setting procedure of the flags.

bp	7	6	5	4	3	2	1	0
Flag	TSnDTLV1	TSnDTLV0	-	-	Reserved	-	TSnDTIE	TSnDTIR
At reset	0	0	-	-	0	-	0	0
Access	R/W	R/W	-	-	R/W	-	R/W	R/W

bp	Flag	Description
7-6	TSnDTLV1-0	Interrupt level flag This flag sets the interrupt level 0 to 3 for interrupt requests.
5-4	-	-
3	Reserved	Always set to “0”.
2	-	-
1	TSnDTIE	Interrupt enable flag 0: Disabled 1: Enabled
0	TSnDTIR	Interrupt request flag 0: No interrupt request has been generated. 1: Interrupt request has been generated.

■ Touch 0, 1 Detect Error Interrupt Control Register (TS0DEICR: 0x03FE8, TS1DEICR: 0x03FEC)

Touch 0, 1 detect error interrupt control register (TS0DEICR, TS1DEICR) controls interrupt level of touch 0, 1 error detect interrupt, interrupt enable and interrupt request. Operate interrupt control register when the maskable interrupt enable flag (MIE) of PSW is "0". Refer to [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup] for the setting procedure of the flags.

bp	7	6	5	4	3	2	1	0
Flag	TSnDELV1	TSnDELV0	-	-	Reserved	-	TSnDEIE	TSnDEIR
At reset	0	0	-	-	0	-	0	0
Access	R/W	R/W	-	-	R/W	-	R/W	R/W

bp	Flag	Description
7-6	TSnDELV1-0	Interrupt level flag This flag sets the interrupt level 0 to 3 for interrupt requests.
5-4	-	-
3	Reserved	Always set to “0”.
2	-	-
1	TSnDEIE	Interrupt enable flag 0: Disabled 1: Enabled
0	TSnDEIR	Interrupt request flag 0: No interrupt request has been generated. 1: Interrupt request has been generated.

■ Touch 0, 1 Cycle Interrupt Control Register (TS0CICR: 0x03FE9, TS1CICR: 0x03FED)

Touch 0, 1 cycle interrupt control register (TS0CICR, TS1CICR) controls interrupt level of touch 0, 1 cycle interrupt, interrupt enable and interrupt request. Operate interrupt control register when the maskable interrupt enable flag (MIE) of PSW is "0". Refer to [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup] for the setting procedure of the flags.

bp	7	6	5	4	3	2	1	0
Flag	TSnCLV1	TSnCLV0	-	-	Reserved	-	TSnCIE	TSnCIR
At reset	0	0	-	-	0	-	0	0
Access	R/W	R/W	-	-	R/W	-	R/W	R/W

bp	Flag	Description
7-6	TSnCLV1-0	Interrupt level flag This flag sets the interrupt level 0 to 3 for interrupt requests.
5-4	-	-
3	Reserved	Always set to “0”.
2	-	-
1	TSnCIE	Interrupt enable flag 0: Disabled 1: Enabled
0	TSnCIR	Interrupt request flag 0: No interrupt request has been generated. 1: Interrupt request has been generated.

■ Touch 0, 1 Data Transmission Interrupt Control Register (TS0ATICR: 0x03FEA, TS1ATICR: 0x03FEE)

Touch 0, 1 transmission interrupt control register (TS0ATICR, TS1ATICR) controls interrupt level of touch 0, 1 transmission interrupt, interrupt enable and interrupt request. Operate interrupt control register when the maskable interrupt enable flag (MIE) of PSW is "0". Refer to [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup] for the setting procedure of the flags.

bp	7	6	5	4	3	2	1	0
Flag	TSnATLV1	TSnATLV0	-	-	Reserved	-	TSnATIE	TSnATIR
At reset	0	0	-	-	0	-	0	0
Access	R/W	R/W	-	-	R/W	-	R/W	R/W

bp	Flag	Description
7-6	TSnATLV1-0	Interrupt level flag This flag sets the interrupt level 0 to 3 for interrupt requests.
5-4	-	-
3	Reserved	Always set to “0”.
2	-	-
1	TSnATIE	Interrupt enable flag 0: Disabled 1: Enabled
0	TSnATIR	Interrupt request flag 0: No interrupt request has been generated. 1: Interrupt request has been generated.

■ Peripheral Function Group Interrupt Level Setting Register (PERIILR: 0x03FFE)

Peripheral Function Group Interrupt Level Setting Register (PERIILR) is used to control interrupt level of peripheral function group interrupt. Use interrupt control register when the Maskable Interrupt Enable flag (MIE) of PSW is "0". Refer to [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup] for the setting procedure of the flags.

bp	7	6	5	4	3	2	1	0
Flag	PERILV1	PERILV0	-	-	Reserved	-	-	-
At reset	0	0	-	-	0	-	-	-
Access	R/W	R/W	-	-	R/W	-	-	-

bp	Flag	Description
7-6	PERILV1-0	Interrupt level specification flag This flag sets the interrupt level 0 to 3 for interrupt requests.
5-4	-	-
3	Reserved	Always set to “0”.
2-0	-	-

Do not change the value of PERIILR when the peripheral function group interrupt factor retention register corresponding to the peripheral function group interrupt input enable register (IRQEXPEN) is set. If changed, an unintended interrupt may be processed.

■ Peripheral Function Group Interrupt Input Enable Register (IRQEXPEN: 0x03F4E)

Peripheral function group interrupt input enable register (IRQEXPEN) is used to control peripheral function group interrupt factors. Use interrupt control register when the Maskabke Interrupt Enable flag (MIE) of PSW is "0". Refer to [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup] for the setting procedure of the flags.

bp	7	6	5	4	3	2	1	0
Flag	Reserved	IRQEXP EN6	IRQEXP EN5	IRQEXP EN4	IRQEXP EN3	IRQEXP EN2	IRQEXP EN1	IRQEXP EN0
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7	Reserved	Always set to "0"
6	IRQEXPEN6	Timer 9 TCMPA Compare Match Interrupt enable flag 0: Disabled 1: Enabled
5	IRQEXPEN5	Serial 4 Stop Condition Interrupt enable flag 0: Disabled 1: Enabled
4	IRQEXPEN4	Serial 4 Interrupt enable flag 0: Disabled 1: Enabled
3	IRQEXPEN3	Serial 2 UART Reception Interrupt enable flag 0: Disabled 1: Enabled
2	IRQEXPEN2	Serial 2 Transmission Interrupt enable flag 0: Disabled 1: Enabled
1	IRQEXPEN1	Serial 1 UART Reception Interrupt enable flag 0: Disabled 1: Enabled
0	IRQEXPEN0	Serial 1 Transmission Interrupt enable flag 0: Disabled 1: Enabled

■ Peripheral Function Group Interrupt Input Factor Retention Register (IRQEXPDT: 0x03F4F)

Peripheral function group interrupt factor retention register (IRQEXPDT) is used to retain the peripheral function group interrupt factors. Use interrupt control register when the Maskabke Interrupt Enable flag (MIE) of PSW is "0". Refer to [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup] for the setting procedure of the flags.

bp	7	6	5	4	3	2	1	0
Flag	Reserved	IRQEXP DT6	IRQEXP DT5	IRQEXP DT4	IRQEXP DT3	IRQEXP DT2	IRQEXP DT1	IRQEXP DT0
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7	Reserved	Always set to "0"
6	IRQEXPDT6	Timer 9 TCMPA Compare Match Interrupt request flag 0: No interrupt request has been generated. 1: An interrupt request has been generated.
5	IRQEXPDT5	Serial 4 Stop Condition Interrupt request flag 0: No interrupt request has been generated. 1: An interrupt request has been generated.
4	IRQEXPDT4	Serial 4 Interrupt request flag 0: No interrupt request has been generated. 1: An interrupt request has been generated.
3	IRQEXPDT3	Serial 2 UART Reception Interrupt request flag 0: No interrupt request has been generated. 1: An interrupt request has been generated.
2	IRQEXPDT2	Serial 2 Transmission Interrupt request flag 0: No interrupt request has been generated. 1: An interrupt request has been generated.
1	IRQEXPDT1	Serial 1 UART Reception Interrupt request flag 0: No interrupt request has been generated. 1: An interrupt request has been generated.
0	IRQEXPDT0	Serial 1 Transmission Interrupt request flag 0: No interrupt request has been generated. 1: An interrupt request has been generated.

4.3 External Interrupts

There are 5 external interrupts in this LSI. The circuit (external interrupt interface), operates the external interrupt input signal, is built-in between the external interrupt input pin and the external interrupt block. This external interrupt interface can manage to do with any kind of external interrupts.

4.3.1 Overview

Table:4.3.1 shows the list of functions which external interrupts 0 to 4/Key interrupt are used.

Table:4.3.1 External Interrupt Functions

Table remarks √: With function -: Without function

	External interrupt input pin	Programmable active edge interrupt	Both edges interrupt	Noise filter	Key input interrupt	Level interrupt	MN101 EFA8	MN101 EFA3	MN101 EFA7	MN101 EFA2
External interrupt 0	P20	√	-	√	-	-	√	√	√	√
External interrupt 1	P21	√	-	√	-	-	√	√	√	√
External interrupt 2	P22	√	√	√	-	√	√	√	√	√
External interrupt 3	P23	√	√	√	-	√	√	√	√	√
External interrupt 4 Key interrupt	P24	√	√	√	-	√	√	√	√	√
	P70 to P77	√	√	-	√	-	√	√	√	√

The pulse which is shorter than 125 ns or 2 cycles of system clock (fs) is neglected because the external interrupts from pins and the key input interrupt events are acknowledged at the rising of the system clock (fs).

Both external interrupt 4 and key interrupt cannot be used at the same time because they share interrupt control register (IRQ4ICR). The KEYT3SEL flag of key interrupt control register 1 (KEYT3_1IMD) is used to select either one to use.

4.3.2 Block Diagram

■ External Interrupt Interface 0 Block Diagram

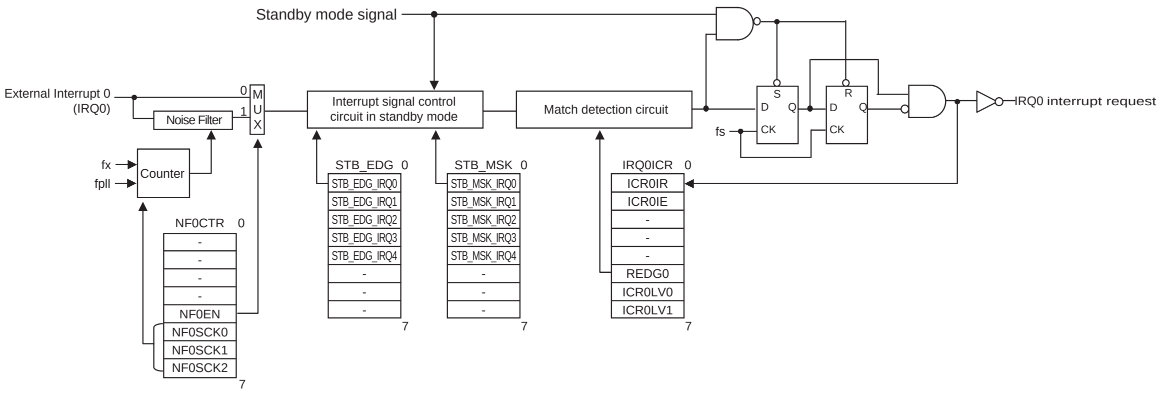


Figure:4.3.1 External Interrupt Interface 0 Block Diagram

■ External Interrupt Interface 1 Block Diagram

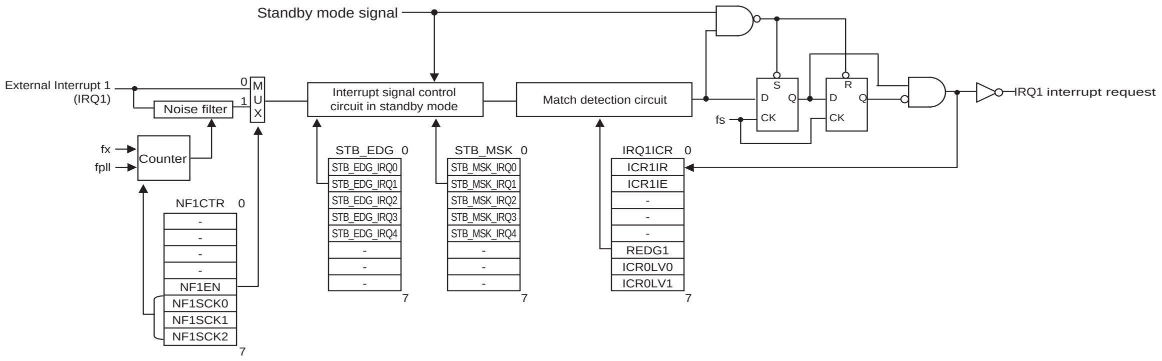


Figure:4.3.2 External Interrupt Interface 1 Block Diagram

■ External Interrupt Interface 2 Block Diagram

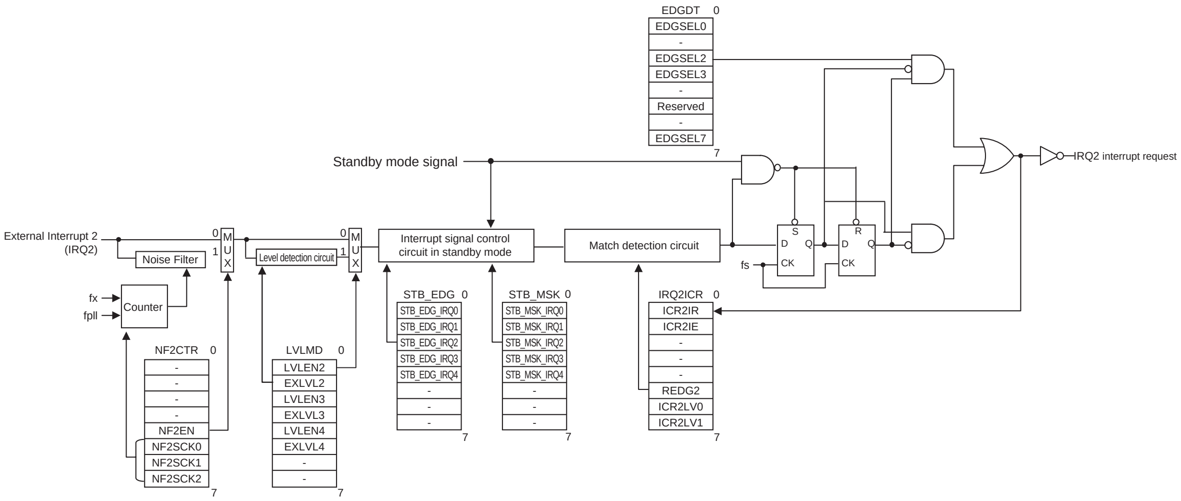


Figure:4.3.3 External Interrupt Interface 2 Block Diagram

■ External Interrupt Interface 3 Block Diagram

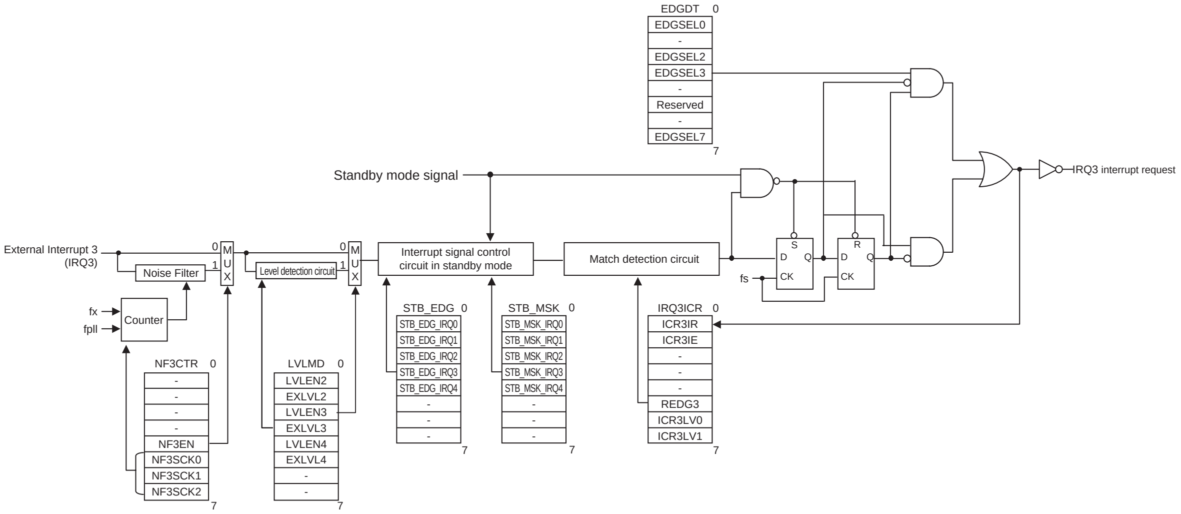


Figure:4.3.4 External Interrupt Interface 3 Block Diagram

■ External Interrupt Interface 4/Key Input Block Diagram

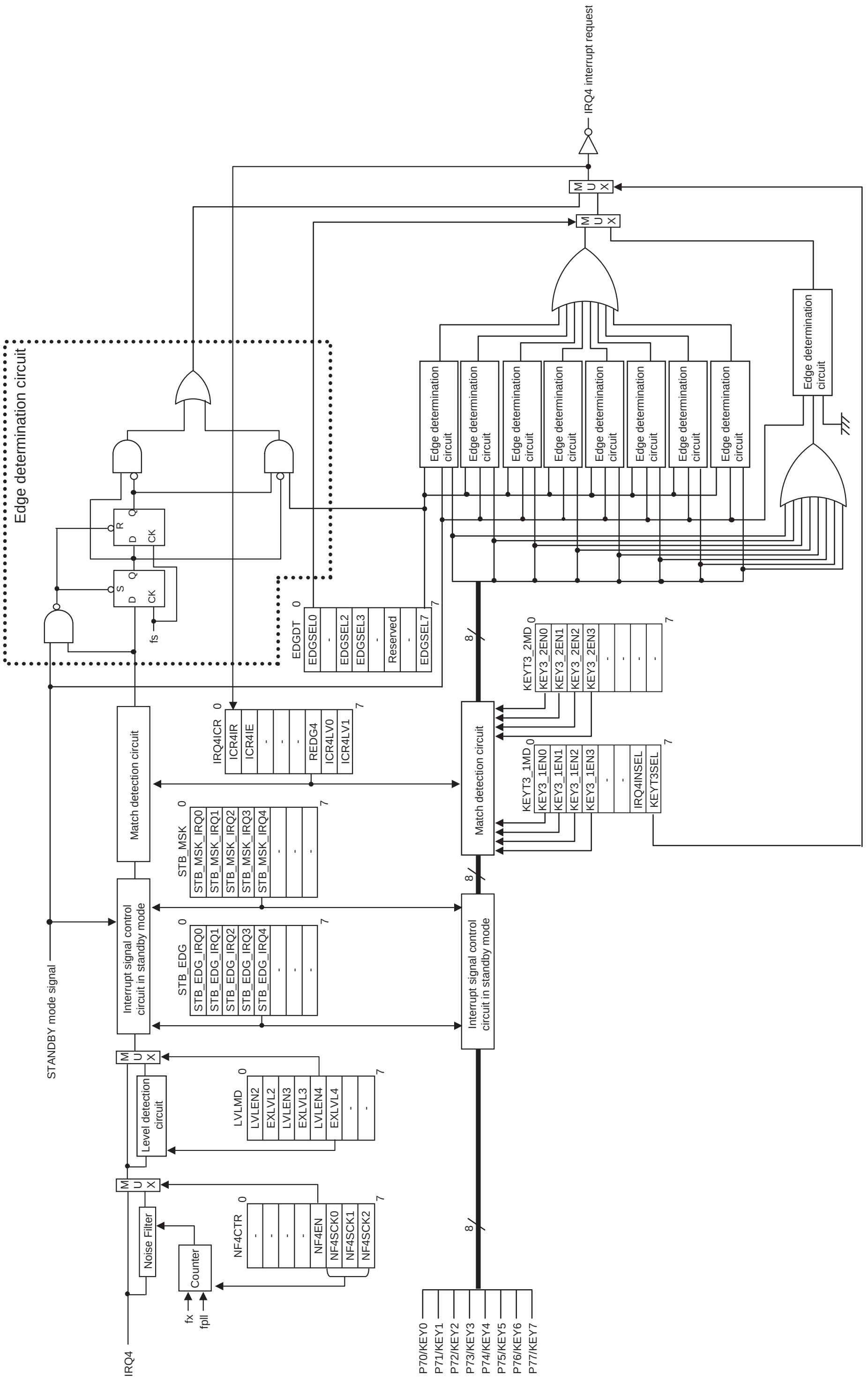


Figure:4.3.5 External Interrupt Interface 4 Block Diagram

4.3.3 External Interrupt Control Registers

External interrupt 0 to 3 and external interrupt 4/Key interrupt generate interrupt requests by the external interrupt input signals passed through each respective external interrupt interface.

External interrupt interfaces 0 to 3 and external interrupt interface 4/Key interrupt interface are controlled by external interrupt n control register (IRQnICR).
External interrupt interface 0 to 1 are controlled by noise filter n control register (NF0CTR and NF1CTR) and external interrupt pin setting register (IRQCNT).
External interrupt interface 2 to 3 are controlled by both edges interrupt control register (EDGDT), external interrupt valid input switch control register (LVLMD), noise filter n control register (NF2CTR and NF3CTR) and IRQCNT register.
External interrupt interface 4/Key interrupt interface are controlled by EDGDT register, LVLMD register and noise filter 4 control register (NF4CTR), IRQCNT register, key interrupt control register 1 (KEYT3_1IMD) and key interrupt control register 2 (KEYT3_2IMD).

In addition, external interrupt interface 0 to 4/Key interrupt interface are used to control interrupt generation factors in CPU OPERATION mode/STANDBY mode by edge interrupt in STANDBY control register (STB_EDG) and interrupt in STANDBY control register (STB_MSK).

The following table shows external interrupt control registers.

Table:4.3.2 External Interrupt Control Register

External interrupt	Register	Address	R/W	Function	Page
External interrupt 0	IRQ0ICR	0x03FE2	R/W	External interrupt 0 control register	IV-21
	IRQCNT	0x03FD0	R/W	External interrupt pin setting register	IV-44
	NF0CTR	0x03FD1	R/W	Noise filter 0 control register	IV-45
	STB_EDG	0x03F5E	R/W	Edge interrupt in STANDBY control register	IV-50
	STB_MSK	0x03F5F	R/W	Interrupt in STANDBY control register	IV-51
External interrupt 1	IRQ1ICR	0x03FE3	R/W	External interrupt 1 control register	IV-21
	IRQCNT	0x03FD0	R/W	External interrupt pin setting register	IV-44
	NF1CTR	0x03FD2	R/W	Noise filter 1 control register	IV-45
	STB_EDG	0x03F5E	R/W	Edge interrupt in STANDBY control register	IV-50
	STB_MSK	0x03F5F	R/W	Interrupt in STANDBY control register	IV-51
External interrupt 2	IRQ2ICR	0x03FE4	R/W	External interrupt 2 control register	IV-21
	IRQCNT	0x03FD0	R/W	External interrupt pin setting register	IV-44
	NF2CTR	0x03FD3	R/W	Noise filter 2 control register	IV-45
	EDGDT	0x03F1E	R/W	Both edges interrupt control register	IV-46
	LVLMD	0x03FD7	R/W	External interrupt valid input switch control register	IV-47
	STB_EDG	0x03F5E	R/W	Edge interrupt in STANDBY control register	IV-50
	STB_MSK	0x03F5F	R/W	Interrupt in STANDBY control register	IV-51
External interrupt 3	IRQ3ICR	0x03FE5	R/W	External interrupt 3 control register	IV-21
	IRQCNT	0x03FD0	R/W	External interrupt pin setting register	IV-44
	NF3CTR	0x03FD4	R/W	Noise filter 3 control register	IV-45
	EDGDT	0x03F1E	R/W	Both edges interrupt control register	IV-46
	LVLMD	0x03FD7	R/W	External interrupt valid input switch control register	IV-47
	STB_EDG	0x03F5E	R/W	Edge interrupt in STANDBY control register	IV-50
	STB_MSK	0x03F5F	R/W	Interrupt in STANDBY control register	IV-51
External interrupt 4 Key interrupt	IRQ4ICR	0x03FE6	R/W	External interrupt 4 control register	IV-21
	IRQCNT	0x03FD0	R/W	External interrupt pin setting register	IV-44
	NF4CTR	0x03FD5	R/W	Noise filter 4 control register	IV-45
	EDGDT	0x03F1E	R/W	Both edges interrupt control register	IV-46
	LVLMD	0x03FD7	R/W	External interrupt valid input switch control register	IV-47
	KEYT3_1IMD	0x03F3E	R/W	Key interrupt control register 1	IV-48
	KEYT3_2IMD	0x03F3F	R/W	Key interrupt control register 2	IV-49
	STB_EDG	0x03F5E	R/W	Edge interrupt in STANDBY control register	IV-50
	STB_MSK	0x03F5F	R/W	Interrupt in STANDBY control register	IV-51

R/W: Readable/ Writable.

■ External Interrupt Pin Setting Register (IRQCNT: 0x03FD0)

External interrupt pin setting register (IRQCNT) is used to enable or disable external interrupt 0 to 4.
Operate this register when the maskable interrupt enable flag (MIE) of PSW is "0".

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	P24EN	P23EN	P22EN	P21EN	P20EN
At reset	-	-	-	0	0	0	0	0
Access	-	-	-	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-5	-	-
4	P24EN	P24 external interrupt 4 input 0: Disabled 1: Enabled
3	P23EN	P23 external interrupt 3 input 0: Disabled 1: Enabled
2	P22EN	P22 external interrupt 2 input 0: Disabled 1: Enabled
1	P21EN	P21 external interrupt 1 input 0: Disabled 1: Enabled
0	P20EN	P20 external interrupt 0 input 0: Disabled 1: Enabled

Change flags in accordance with step (2) in [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup].

Set P2nEN flag to “0” to fix the input of external interrupt to “0” except when P2IN register is being read.

To enable maskable interrupts after setting P2nEN flag to “1”, make sure to set IR flag of the corresponding interrupt to “0” beforehand.

When external interrupt is not used, set IRQnIE flag to “0”.
Unintended interrupt may be generated when Port 2 input register (P2IN) is read while the IRQnIE flag is set to “1”, even if the P2nEN flag is set to “0”.

The interrupt request flag may have been set when the interrupt edge is switched. It is necessary to specify the interrupt valid edge before interrupts are enabled. Also, if the interrupt request flag may have been already set, it must be cleared.

■ Noise Filter 0 to 4 Control Registers (NF0CTR: 0x03FD1, NF1CTR: 0x03FD2, NF2CTR: 0x03FD3, NF3CTR: 0x03FD4, NF4CTR: 0x03FD5)

Noise filter 0 to 4 control registers (NF0CTR to NF4CTR) are used to set the noise removal function on external interrupt 0 to 4. They are also used to select the sampling cycle of noise removal function.
Operate NFnEN1 flags when the maskable interrupt enable flag (MIE) of PSW is "0".

bp	7	6	5	4	3	2	1	0
Flag	NFnSCK2	NFnSCK1	NFnSCK0	NFnEN1	-	-	-	-
At reset	0	0	0	0	-	-	-	-
Access	R/W	R/W	R/W	R/W	-	-	-	-

bp	Flag	Description
7-5	NFnSCK2-0	IRQn noise sampling frequency 000: fpll 001: fpll/2^{5} 010: fpll/2^{6} 011: fpll/2^{7} 100: fpll/2^{8} 101: fpll/2^{9} 110: fpll/2^{10} 111: fx
4	NFnEN1	Noise filter enable control 0: Disabled 1: Enabled
3-0	-	-

Change flags in accordance with step (2) in [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup].

To enable maskable interrupts after setting NFnEN1 flag to “1”, make sure to set IR flag of the corresponding interrupt to “0” beforehand.

■ Both Edges Interrupt Control Register (EDGDT: 0x03F1E)

Both edges interrupt control register (EDGDT) is used to select interrupt edges of IRQ2 to 4. With this register, the edge to generate interrupts is selected; both edges or the edge specified by external interrupt control register (IRQ2ICR, IRQ3ICR, IRQ4ICR). EDGDT register is also used to select bit common or bit independent for the edge determination circuit in the key interrupt block.

bp	7	6	5	4	3	2	1	0
Flag	EDGSEL7	-	-	-	EDGSEL3	EDGSEL2	-	EDGSEL0
At reset	0	-	-	-	0	0	-	0
Access	R/W	-	-	-	R/W	R/W	-	R/W

bp	Flag	Description
7	EDGSEL7	IRQ4 both edges operation selection 0: Programmable active edge interrupt (Specified by REDG4 of IRQ4ICR) 1: Both edges interrupt
6-4	-	-
3	EDGSEL3	IRQ3 both edges operation selection 0: Programmable active edge interrupt (Specified by REDG3 of IRQ3ICR) 1: Both edges interrupt
2	EDGSEL2	IRQ2 both edges operation selection 0: Programmable active edge interrupt (Specified by REDG2 of IRQ2ICR) 1: Both edges interrupt
1	-	-
0	EDGSEL0	Key interrupt selection 0: Key input edge detection circuit bit common * 1: Key input edge detection circuit bit independent

Change flags in accordance with step (2) in [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup].
* Refer to [Chapter 4 4.3.7 Key Input Interrupt] for details.

When “Key input edge detection circuit bit common” is selected by the EDGSEL0 flag, an interrupt request signal is generated by inputting OR of the match detection circuit output for the valid key input. During the match detection by a single valid key input, no interrupt request signal is generated even if a match is detected by another valid key input.

When “Key input edge detection circuit bit independent” is selected by the EDGSEL0 flag, the edge determination circuit is used for each valid key input. Thus, even during match detection by a single valid key input, an interrupt request signal is generated by detecting a match by another valid key input.

■ External Interrupt Valid Input Switch Control Register (LVLMD: 0x03FD7)

bp	7	6	5	4	3	2	1	0
Flag	-	-	EXLVL4	LVLEN4	EXLVL3	LVLEN3	EXLVL2	LVLEN2
At reset	-	-	0	0	0	0	0	0
Access	-	-	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-6	-	-
5	EXLVL4	External interrupt 4 valid input level setting (Enabled at LVLEN4="1") 0: Low-level 1: High-level
4	LVLEN4	External interrupt 4 valid input setting 0: Edge 1: Level
3	EXLVL3	External interrupt 3 valid input level setting (Enabled at LVLEN3="1") 0: Low-level 1: High-level
2	LVLEN3	External interrupt 3 valid input setting 0: Edge 1: Level
1	EXLVL2	External interrupt 2 valid input level setting (Enabled at LVLEN2="1") 0: Low-level 1: High-level
0	LVLEN2	External interrupt 2 valid input setting 0: Edge 1: Level

Change flags in accordance with step (2) in [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup].

■ Key Interrupt Control Register 1 (KEYT3_1IMD: 0x03F3E)

Key interrupt control register 1 (KEYT3_1IMD) is used to select whether key interrupt or external interrupt 4 is enabled. This register can also select which pin of Port A should be used to enable key interrupt for each bit. Operate KEYT3_1ENn flags when the maskable interrupt enable flag (MIE) of PSW is "0".

bp	7	6	5	4	3	2	1	0
Flag	KEYT3 SEL	-	-	-	KEYT3_1 EN3	KEYT3_1 EN2	KEYT3_1 EN1	KEYT3_1 EN0
At reset	0	-	-	-	0	0	0	0
Access	R/W	-	-	-	R/W	R/W	R/W	R/W

bp	Flag	Description
7	KEYT3SEL	Interrupt source selection 0: External interrupt 4 1: Key interrupt
6-4	-	-
3	KEYT3_1EN3	KEY3 interrupt selection 0: Disabled 1: Enabled
2	KEYT3_1EN2	KEY2 interrupt selection 0: Disabled 1: Enabled
1	KEYT3_1EN1	KEY1 interrupt selection 0: Disabled 1: Enabled
0	KEYT3_1EN0	KEY0 interrupt selection 0: Disabled 1: Enabled

Change flags in accordance with step (2) in [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup].

Both external interrupt 4 and key interrupt cannot be used at the same time because they share interrupt control register (IRQ4ICR). The KEYT3SEL flag of key interrupt control register 1 (KEYT3_1IMD) is used to select either one to use.

To enable maskable interrupts after setting KEYT3_1ENn flag to “1”, make sure to set IR flag of the corresponding interrupt to “0” beforehand.

■ Key Interrupt Control Register 2 (KEYT3_2IMD: 0x03F3F)

Key interrupt control register 2 (KEYT3_2IMD) is used to select which pin of Port A should be used to enable key interrupt for each bit.
Operate KEYT3_2IMD register when the maskable interrupt enable flag (MIE) of PSW is "0".

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	KEYT3_2EN3	KEYT3_2EN2	KEYT3_2EN1	KEYT3_2EN0
At reset	-	-	-	-	0	0	0	0
Access	-	-	-	-	R/W	R/W	R/W	R/W

bp	Flag	Description
7-4	-	-
3	KEYT3_2EN3	KEY7 interrupt selection 0: Disabled 1: Enabled
2	KEYT3_2EN2	KEY6 interrupt selection 0: Disabled 1: Enabled
1	KEYT3_2EN1	KEY5 interrupt selection 0: Disabled 1: Enabled
0	KEYT3_2EN0	KEY4 interrupt selection 0: Disabled 1: Enabled

To enable maskable interrupts after setting KEYT3_2ENn flag to “1”, make sure to set IR flag of the corresponding interrupt to “0” beforehand.

■ Edge Interrupt in STANDBY Control Register (STB_EDG: 0x03F5E)

Edge interrupt in STANDBY control register (STB_EDG) is used to select the factors of external interrupt 0 to 4/ key interrupt generation in STANDBY mode between edge detection and level detection of input signal.

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	STB_EDG _IRQ4	STB_EDG _IRQ3	STB_EDG _IRQ2	STB_EDG _IRQ1	STB_EDG _IRQ0
At reset	-	-	-	0	0	0	0	0
Access	-	-	-	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-5	-	-
4	STB_EDG _IRQ4	External interrupt 4/Key interrupt Edge interrupt operation setting in STANDBY mode 0: Level interrupt 1: Edge interrupt
3	STB_EDG _IRQ3	External interrupt 3 Edge interrupt operation setting in STANDBY mode 0: Level interrupt 1: Edge interrupt
2	STB_EDG _IRQ2	External interrupt 2 Edge interrupt operation setting in STANDBY mode 0: Level interrupt 1: Edge interrupt
1	STB_EDG _IRQ1	External interrupt 1 Edge interrupt operation setting in STANDBY mode 0: Level interrupt 1: Edge interrupt
0	STB_EDG _IRQ0	External interrupt 0 Edge interrupt operation setting in STANDBY mode 0: Level interrupt 1: Edge interrupt

It is necessary to clear STB_EDG register and STB_MSK register to “00” before changing the value of IRQCNT register, NFnCTR register and LVLMD register.

Settings "Edge interrupt" by the STB_EDG_IRQn flag of STB_EDG register and "Level" by the LVLENn flag of LVLMD register cannot be used at the same time. If using them together, unintended interrupt may be generated.

■ Interrupt in STANDBY Control Register (STB_MSK: 0x03F5F)

Interrupt in STANDBY control register (STB_MSK) is used to select the execution timing of interrupt between before STANDBY mode transition and after STANDBY mode transition, when IRQn interrupt signal generates just before the transition to STANDBY mode.

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	STB_MSK _IRQ4	STB_MSK _IRQ3	STB_MSK _IRQ2	STB_MSK _IRQ1	STB_MSK _IRQ0
At reset	-	-	-	0	0	0	0	0
Access	-	-	-	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-5	-	-
4	STB_MSK _IRQ4	IRQ4 interrupt execution timing control until the transition to STANDBY mode 0: Normal operation (Before STANDBY mode transition) 1: Masking operation (After STANDBY mode transition)
3	STB_MSK _IRQ3	IRQ3 interrupt execution timing control until the transition to STANDBY mode 0: Normal operation (Before STANDBY mode transition) 1: Masking operation (After STANDBY mode transition)
2	STB_MSK _IRQ2	IRQ2 interrupt execution timing control until the transition to STANDBY mode 0: Normal operation (Before STANDBY mode transition) 1: Masking operation (After STANDBY mode transition)
1	STB_MSK _IRQ1	IRQ1 interrupt execution timing control until the transition to STANDBY mode 0: Normal operation (Before STANDBY mode transition) 1: Masking operation (After STANDBY mode transition)
0	STB_MSK _IRQ0	IRQ0 interrupt execution timing control until the transition to STANDBY mode 0: Normal operation (Before STANDBY mode transition) 1: Masking operation (After STANDBY mode transition)

It is necessary to clear STB_EDG register and STB_MSK register to “00” before changing the value of IRQCNT register, NFnCTR register and LVLMD register.

Settings "Masking operation" by the STB_MSK_IRQn flag of STB_MSK register and "Level" by the LVLENn flag of LVLMD register cannot be used at the same time. If using them together, unintended interrupt may be generated.

If "Masking operation" is selected and LSI returns from STANDBY mode by external interrupt IRQn (including retention request) generated after the transition to STANDBY mode, the STB_MSK_IRQn flag is automatically turned to "0".

4.3.4 Programmable Active Edge Interrupt

■ Programmable Active Edge Interrupt (External Interrupt 0 to 4/Key interrupt)

Programmable active edge interrupt is the function which can select rising or falling edge for the signal input from the external interrupt input pin, then generate interrupts at the selected edge.

For external interrupt 0 to 4/Key interrupt in STANDBY mode, the interrupt generation factor differs depending on the value of STB_EDG and STB_MSK registers.
[Chapter 4 4.3.8 External Interrupt Operation in STANDBY Mode].

■ Programmable Active Edge Interrupt Setup Example (External Interrupt 0 to 4/Key interrupt)

External interrupt 0 (IRQ0) is generated at rising edge of the input signal from pin P20.

The table below shows a setup example.

Setup Procedure	Description
(1) Set the external interrupt IRQCNT (0x03FD0) bp0: P20EN = 1	(1) Set the P20EN flag of IRQCNT register to “1” to set P20 as an external interrupt.
(2) Specify the interrupt active edge IRQ0ICR (0x03FE2) bp5: REDG0 = 1	(2) Set the REDG0 flag of IRQ0ICR register to “1” to specify rising edge as the active edge for interrupts.
(3) Set the interrupt level IRQ0ICR (0x03FE2) bp7 to 6: IRQ0LV1 to 0 = 10	(3) Set the interrupt priority level in the IRQ0LV1 to 0 flag of IRQ0ICR register. If the IRQ0IR flag has already been set, it must be cleared. [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup]
(4) Enable interrupts IRQ0ICR (0x03FE2) bp1: IRQ0IE = 1	(4) Set the IRQ0IE flag of IRQ0ICR register to “1” to enable interrupts.

The interrupt request flag may have been set when the interrupt edge is switched. It is necessary to specify the interrupt valid edge before interrupts are enabled. Also, if the IRQ0IR flag has already been set, it must be cleared.

To use the external interrupt pin as interrupt function, it needs to be pulled up in advance to prevent intermediate potential input.

4.3.5 Both Edges Interrupt

■ Both Edges Interrupt (External Interrupts 2 to 4/Key interrupt)

Both edges interrupt is the function which can generates interrupt at both falling and rising edges for the signal input from the external interrupt input pin.

For external interrupt 0 to 4/Key interrupt in STANDBY mode, the interrupt generation factor differs depending on the value of STB_EDG and STB_MSK registers.
[Chapter 4 4.3.8 External Interrupt Operation in STANDBY Mode].

■ Both Edges Interrupt Setup Example (External Interrupts 2 to 4/Key interrupt)

External interrupt 2 (IRQ2) is generated at the both edges of the input signal from pin P22.
The table below shows a setup example.

Setup Procedure	Description
(1) Set the external interrupt IRQCNT (0x03FD0) bp2: P22EN = 1	(1) Set the P22EN flag of IRQCNT register to “1” to set P22 to an external interrupt.
(2) Select the both edges interrupt EDGDT (0x03F1E) bp2: EDGSEL2 = 1	(2) Set the EDGSEL2 flag of EDGDT register to "1" to select the both edges interrupt.
(3) Set the interrupt level IRQ2ICR (0x03FE4) bp7 to 6: IRQ2LV1 to 0 = 10	(3) Set the interrupt level by the IRQ2LV1 to 0 flag of IRQ2ICR register. If the IRQ2IR flag has already been set, it must be cleared. [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup]
(4) Enable the interrupt IRQ2ICR (0x03FE4) bp1: IRQ2IE = 1	(4) Set the IRQ2E flag of IRQ2ICR register to "1" to enable the interrupt.

When the both edges interrupt is selected, the interrupt request is generated at both edge, regardless of the REDGn flag of IRQnICR register.

The interrupt request flag may have been set when the interrupt edge is switched. It is necessary to clear the interrupt request flag before interrupts are enabled. Also, the both edges interrupt must be selected before interrupts are enabled.

To use the external interrupt pin as interrupt function, it needs to be pulled up in advance to prevent intermediate potential input.

4.3.6 Level Interrupt

■ Level Interrupt (External interrupt 2 to 4)

Level interrupt is the function which can select the input level High or input level Low for the signal input from the external interrupt input pin, then generate interrupts at the selected level. This interrupt can return LSI from STANDBY mode.

■ Level Interrupt Example (External interrupt 2 to 4)

External interrupt 2 (IRQ2) is generated at "High" level of the input signal from pin P22.
The table below shows a setup example of IRQ2.

Setup Procedure	Description
(1) Set external interrupt IRQCNT (0x03FD0) bp2: P22EN = 1	(1) Set the P22EN flag of IRQCNT register to “1” to set P22 to an external interrupt.
(2) Specify the interrupt valid edge IRQ2ICR (0x03FE4) bp5: REDG2 = 1	(2) Set the REDG2 flag of IRQ2ICR register to “0” and specify the rising edge as the valid edge.
(3) Specify the interrupt valid input LVLMD (0x03FD7) bp1: EXLVL2 = 1	(3) Set the EXLVL flag of LVLMD register to “1” to specify “High-level” as the interrupt valid input level.
(4) Enable the level interrupt LVLMD (0x03FD7) bp0: LEVEN2 = 1	(4) Set the LEVEN2 flag of LVLMD register to “1” to specify “Level interrupt” as the interrupt valid input.
(5) Set the interrupt level IRQ2ICR (0x03FE4) bp7 to 6: IRQ2LV1 to 0 = 10	(5) Set the interrupt priority level in the IRQ2LV1 to 0 flag of IRQ2ICR register. If the IRQ2IR flag has already been set, it must be cleared. [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup]
(6) Enable the interrupt IRQ2ICR (0x03FE4) bp1: IRQ2IE = 1	(6) Set the IRQ2IE flag of IRQ2ICR register to “1” enable the interrupt.

External interrupt 2 is generated at the "High" level of the input signal from P22.

Set external interrupt valid input level equal to the polarity of interrupt active edge.
When external interrupt valid input level is High-level, interrupt active edge is rising edge.
When external interrupt valid input level is Low-level, interrupt active edge is falling edge.

The interrupt request flag may have been set when the interrupt edge is switched.
The interrupt active edge must be specified before interrupts are enabled.

! When the level interrupt function is used, an interrupt may be generated again after completion of the interrupt process program.
When the level interrupt function is used, disable the external interrupt enable flag in the interrupt process program.

! Settings "Edge interrupt" by the STB_EDG_IRQn flag of STB_EDG register and "Level" by the LVLENn flag of LVLMD register cannot be used at the same time. If using them together, unintended interrupt may be generated.

! Settings "Masking operation" by the STB_MSK_IRQn flag of STB_MSK register and "Level" by the LVLENn flag of LVLMD register cannot be used at the same time. If using them together, unintended interrupt may be generated.

4.3.7 Key Input Interrupt

■ Key Input Interrupt

This LSI can set the pins of Port A (P70 to P77) as the key input pins for each bit. Key input interrupt is the function which can generate interrupts at the falling edge if at least one key input pin outputs "Low" level.
Also, when rising edge is selected by REDG4 flag of ICR4ICR register, the interrupts can be generated at the rising edge if at least one key input pin is a "High" level.

For external interrupt 0 to 4/Key interrupt in STANDBY mode, the interrupt generation factor differs depending on the value of STB_EDG and STB_MSK registers.
[Chapter 4 4.3.8 External Interrupt Operation in STANDBY Mode].

Key input pins need to be pulled up in advance to prevent intermediate potential input.

■ Key Input Interrupt Setup Example

After setting pins P70 to P73 of Port A as the key input pins, external interrupt 4 (IRQ4) is generated when the key is input (Low-level).

An example setup procedure, with a description of each step is shown below.

Setup Procedure	Description
(1) Set the key input to input P7DIR(0x03E97) bp7 to 0: P7DIR7 to 0 =00000000	(1) Set the P7DIR7 to 0 flags of P7DIR register to "00000000" and set pins P70 to P73 to the input pin
(2) Set the pull to up resistor P7PLU(0x03EA7) bp7 to 0: P7PLU7 to 0 =00001111	(2) Set the P7PLU7 to 0 flags of P7PLU register to "00001111" to add pull-up resistors to pins P70 to P73.
(3) Select the key input interrupt KEYT3_1IMD (0x03F3E) bp7: KEYT3_1SEL =1	(3) Set the KEYT3SEL flag of KEYT3_1IMD register to "1" to select the key interrupt as interrupt source.
(4) Select the key input pin KEYT3_1IMD(0x03F3E) bp3 to 0: KEYT3_1EN3 to 0 =1111	(4) Set the KEYT3_1EN3 to 0 flags of KEYT3_1IMD register to "1111" to set pins P70 to P73 as the key input pins.
(5) Specify the interrupt active edge EDGDT(0x03F1E) bp7: EDGSEL7 =0 bp0: EDGSEL0 =0	(5) Set the EDGSEL7 and 0 flags of EDGDT register to “0” to set the interrupt edge operation to the interrupt active edge.
IRQ4ICR(0x03FE6) bp5: REDG0 =0	Set the REDG0 flag of IRQ4ICR register to “0” to specify falling edge as the interrupt active edge.
(6) Set the interrupt level IRQ4ICR(0x03FE6) bp7 to 6: IRQ4LV1 to 0 =10	(6) Set the interrupt level by the IRQ4LV1 to 0 flag of the IRQ4ICR register. If the interrupt request flag has been already set, clear the request flag (IRQ4IR). [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup]
(7) Enable interrupts IRQ4ICR(0x03FE6) bp1: IRQ4IE =1	(7) Set the IRQ4IE flag of the IRQ4ICR register to "1" to enable the interrupt.

*Above (3) and (4) can be set at the same time.

If there is at least one signal, which is input from the P70 to P73 pins, shows "Low" level, the external interrupt 4 is generated at the falling edge.

The key input must be set before interrupts are enabled.

When “Key input edge detection circuit bit common” is selected by the EDGSEL0 flag, an interrupt request signal is generated by inputting OR of the match detection circuit output for the valid key input. During the match detection by a single valid key input, no interrupt request signal is generated even if a match is detected by another valid key input.

If selecting the both edge interrupt by the EDGSEL7 flag of EDGDT register, the following edge key interrupts are generated.
(When EDGSEL0 flag = 0: key input edge detection circuit bit common)
OR of the enable key input match detection circuit is determined with both edges and interrupt request signal is generated.
(When EDGSEL0 flag = 1: key input edge detection circuit bit independent)
The both edge determination is executed for the match detection circuit output every valid key input and interrupt request signal is generated.
* Match detection circuit outputs “High” by external interrupt 4 control register (REDG4 flag of IRQ4ICR register) under the following condition;
In falling edge setup: “Low” level match
In rising edge setup: “High” level match

4.3.8 External Interrupt Operation in STANDBY Mode

This LSI can return from STANDBY mode (HALT0/HALT2/STOP0) with an interrupt from external interrupt input 0 to 4/Key interrupt. The interrupt generation factor of external interrupt 0 to 4/Key interrupt, differs between in CPU OPERATION mode and STANDBY mode.
For external interrupt 0 to 4/Key interrupt in STANDBY mode, the interrupt generation can be selected by the STB_EDG_IRQn flag of STB_EDG register and the STB_MSK_IRQn flag of STB_MSK register.

For CPU OPERATION mode and STANDBY mode, refer to [Chapter 3 Clock Control].

The external interrupt generation factors in CPU OPERATION mode/STANDBY mode depending on the combination with the STB_EDG_IRQn flag, the STB_MSK_IRQn flag and the LVLENn flag of LVLMD register.

Table:4.3.3 External Interrupt Generation Factors Around Transition to STANDBY Mode

External interrupt control flag			Interrupt generation factor	
STB_EDG_IRQn	STB_MSK_IRQn	LVLENn	CPU OPERATION	STANDBY
0	0	0	Edge interrupt	Level interrupt
0	0	1	Level interrupt	Level interrupt
1	0	0	Edge interrupt	Edge interrupt
0	1	0	Masking operation	Edge interrupt
1	1	0	Masking operation	Edge interrupt
Other combination			Setting prohibited	

Settings "Edge interrupt" by the STB_EDG_IRQn flag of STB_EDG register and "Level" by the LVLENn flag of LVLMD register cannot be used at the same time. If using them together, unintended interrupt may be generated.

Settings "Masking operation" by the STB_MSK_IRQn flag of STB_MSK register and "Level" by the LVLENn flag of LVLMD register cannot be used at the same time. If using them together, unintended interrupt may be generated.

■ Level Interrupt in STANDBY Mode

When "Level interrupt" is selected as the interrupt generation factor of external interrupt 0 to 4/Key interrupt, an interrupt is generated if the external interrupt input reaches the level set by the REDGn flag of IRQnICR register. When [falling edge ("Low" level)] is selected by the REDGn flag, an interrupt is generated if the interrupt input reaches "Low" level in STANDBY mode. If the interrupt input has been at "Low" level before the transition to STANDBY mode, an interrupt is generated right after the transition and the LSI returns from STANDBY mode. When [rising edge ("High" level)] is selected by the REDGn flag, an interrupt is generated if the interrupt input reaches "High" level in STANDBY mode. If the interrupt input has been at "High" level before the transition to STANDBY mode, an interrupt is generated right after the transition and the LSI returns from STANDBY mode.

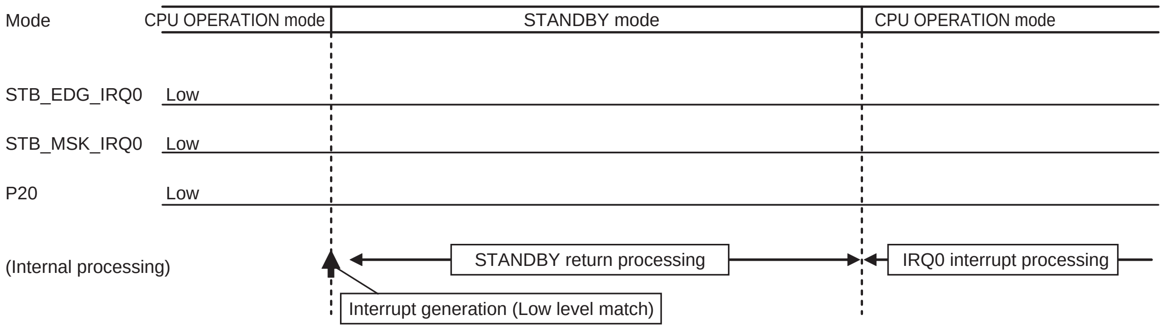


Figure:4.3.6 Timing 1 of P20 Interrupt Generation during STANDBY Mode
Level Interrupt (LVLEN0 = 0, REDG0 = 0)

For level interrupt, regardless of the value set to the EDGSELn flag of EDGDT register, an interrupt is generated when the interrupt input reaches the level set by the REDGn flag of IRQnICR register.

■ Edge Interrupt in STANDBY Mode

When "Edge interrupt" is selected as the interrupt generation factor of external interrupt 0 to 4/Key interrupt, an interrupt is generated if the external interrupt input reaches the edge set by the REDGn flag of IRQnICR register and the EDGSELn flag of EDGDT register.

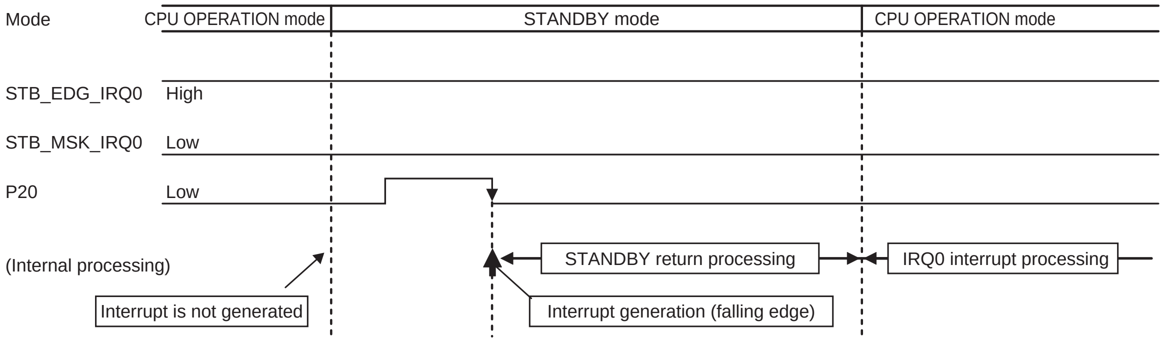

Mode
CPU OPERATION mode
STANDBY mode
CPU OPERATION mode
STB_EDG_IRQ0
High
STB_MSK_IRQ0
Low
P20
Low
(Internal processing)
STANDBY return processing
IRQ0 interrupt processing
Interrupt is not generated
Interrupt generation (falling edge)

Figure:4.3.7 Timing 2 of P20 Interrupt Generation during STANDBY Mode
Edge Interrupt (LVLEN0 = 0, REDG0 = 0)

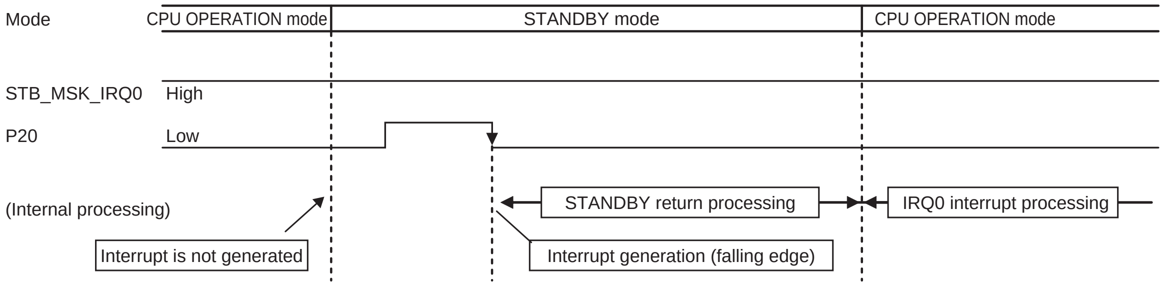

Mode
CPU OPERATION mode
STANDBY mode
CPU OPERATION mode
STB_MSK_IRQ0
High
P20
Low
(Internal processing)
STANDBY return processing
IRQ0 interrupt processing
Interrupt is not generated
Interrupt generation (falling edge)

Figure:4.3.8 Timing 3 of P20 Interrupt Generation during STANDBY Mode
Edge Interrupt (LVLEN0 = 0, REDG0 = 0)

For edge interrupt, an interrupt is generated on the edge set by the REDGn flag of IRQnICR register and the EDGESELn flag of EDGDT register.

When "Masking operation" is selected by the STB_MSK_IRQn flag of STB_MSK register, regardless of the setting value of the STB_EDG_IRQn flag of STB_EDG register, the interrupt generation factor in STANDBY mode will be "Edge interrupt".

When the STB_MSK_IRQn flag of STB_MSK register is set and the LSI returns from STANDBY mode by the external interrupt IRQn generation (including retention request), the STB_MSK_IRQn flag is automatically turned to “0”.

When the STB_MSK_IRQn flag of STB_MSK register is set and the LSI returns from STANDBY mode by the external interrupt IRQn generation (including external interrupts except IRQn), the STB_MSK_IRQn flag remains set but the mask operation is disabled. It is necessary to clear all the flags of STB_MSK register after return from STANDBY mode.

■ Masking Operation in CPU OPERATION Mode

When "Masking operation" is selected as the interrupt generation factor of external interrupt 0 to 4/Key interrupt, no interrupt is generated even if the external interrupt input reaches the edge set by the REDGn flag of IRQnICR register and the EDGSELn flag of EDGDT register. However, the interrupt request at that time is retained in the internal circuit (request retention).
If a request retention is generated before the transition to STANDBY mode, an interrupt is generated by the interrupt request retained after the transition to STANDBY mode and the LSI returns from STANDBY mode.

This function can prevent STANDBY mode return factors generation before the transition to STANDBY mode when the external interrupt 0 to 4/Key interrupt are used as return factors from STANDBY mode.

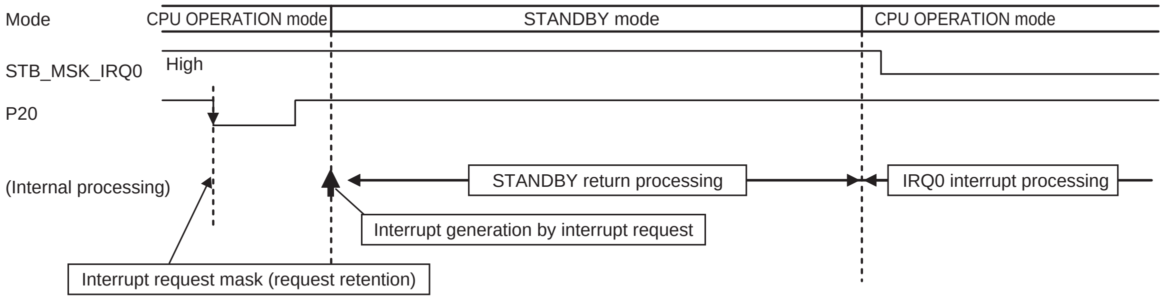

Mode
CPU OPERATION mode
STANDBY mode
CPU OPERATION mode
STB_MSK_IRQ0
High
P20
(Internal processing)
STANDBY return processing
IRQ0 interrupt processing
Interrupt generation by interrupt request
Interrupt request mask (request retention)

Figure:4.3.9 Timing 4 of P20 Interrupt Generation during STANDBY Mode
Masking Operation (LVLEN0=0, REDG0=0)

When the STB_MSK_IRQn flag of STB_MSK register is set and the LSI returns from STANDBY mode by the external interrupt IRQn generation (including retention request) after the transition to STANDBY mode, the STB_MSK_IRQn flag is automatically turned to “0”.

When the STB_MSK_IRQn flag of STB_MSK register is set and the LSI returns from STANDBY mode by the external interrupt IRQn generation (including external interrupts except IRQn), the STB_MSK_IRQn flag remains set but the mask operation is disabled. It is necessary to clear all the flags of STB_MSK register after return from STANDBY mode.

■ External Interrupt Operation in STANDBY Mode (External Interrupt 0 to 4/Key Interrupt)

The LSI can return from STANDBY mode by external interrupts.
The interrupt in STANDBY mode can be selected from either edge interrupt or level interrupt by STB_EDG register.

When the level interrupt is selected during STANDBY mode, an interrupt is generated when the level of the external interrupt pin reaches the value set in the external interrupt valid edge specification flag. Therefore, pay attention to the value set in the external interrupt valid edge specification flag and the level of the corresponding external interrupt pin during the transition to STANDBY mode. When the value of the external interrupt valid edge specification flag reaches the corresponding external interrupt, the LSI returns from STANDBY mode immediately.

■ Return by External Interrupt in STANDBY Mode Setup Example 1

The LSI returns from STOP mode by the external interrupt 0 (IRQ0) generated by the "Low" level signal input from the external interrupt 0 pin.

An example setup procedure, with a description of each step is shown below.

Setup Procedure	Description
(1) Specify interrupt valid edge IRQ0ICR(0x03FE2) bp5: REDG0=0	(1) Set the REDG0 of IRQ0ICR register to “0” to specify falling edge as the interrupt active edge.
(2) Set the external interrupt pin Pull-up the external interrupt 0 pin in advance.	(2) The value of the REDG0 flag of IRQ0ICR register differs from the level of the external interrupt 0 pin.
(3) Set the external interrupt IRQCNT(0x03FD0) bp0: P20EN=1	(3) Set the P20EN flag of IRQCNT register to “1” to set P20 as the external interrupt.
(4) Set the interrupt level IRQ0ICR(0x03FE2) bp7 to 6: IRQ0LV1 to 0=10	(4) Set the interrupt level by setting the IRQ0LV1 to 0 flags of IRQ0ICR register. If IRQ0IR flag may have been already set, make sure to clear it.
(5) Enable interrupts IRQ0ICR(0x03FE2) bp1: IRQ0IE=1	(5) Set the IRQ0IE flag of IRQ0ICR register to “1” to enable interrupts.
(6) Set STOP mode CPUM(0x03F00) bp3: STOP=1	(6) Set the STOP flag of CPUM register to “1” to transit to STOP mode. [Chapter 3 3.3.3 STANDBY Mode]

If the "Low" level signal is input to the external interrupt 0 pin, then, the value of the external interrupt valid edge specification flag (REDG0) and the level of external interrupt 0 pin are matched, the external interrupt 0 is accepted and LSI returns from STOP mode.

The LSI returns from STOP mode after oscillation stabilization wait time set by DLYCTR register has passed since the acceptance of an external interrupt.
[Chapter 2 2.6.2 Oscillation Stabilization Wait time]

■ External Edge Interrupt Operation in STANDBY Mode (External Interrupt 0 to 4/Key Interrupt)

The LSI can return from STANDBY mode by external interrupts.
The interrupt in STANDBY mode can be selected from either edge interrupt or level interrupt by STB_EDG register.

When the edge interrupt is selected, also in STANDBY mode, an interrupt is generated at the edge set in the external interrupt valid edge specification flag. Therefore, even if the value of the external interrupt valid edge specification flag reaches the corresponding external interrupt, the LSI does not return from STANDBY mode and an interrupt is generated by the active edge input after that. When the both edges interrupt is selected by EDGDT register, an interrupt is generated at the both edges of the external interrupt input signal.

■ Return by External Interrupt in STANDBY Mode Setup Example 2

The LSI returns from STOP mode by external interrupt 0 (IRQ0) generated by the falling edge input from pin IRQ0.

An example setup procedure, with a description of each step is shown below.

Setup Procedure	Description
(1) Specify interrupt valid edge IRQ0ICR(0x03FE2) bp5: REDG0=0	(1) Set the REDG0 of IRQ0ICR register to “0” to specify falling edge as the interrupt active edge.
(2) Set the external interrupt pin Pull up the external interrupt 0 pin in advance.	(2) The value of the REDG0 flag of the IRQ0ICR register differs from the level of pin IRQ0.
(3) Set the external interrupt IRQCNT(0x03FD0) bp0: P20EN=1	(3) Set the P20EN flag of IRQCNT register to “1” to set P20 as the external interrupt.
(4) Set the interrupt level IRQ0ICR(0x03FE2) bp7 to 6: IRQ0LV1 to 0=10	(4) Set the interrupt level by the IRQ0LV1 to 0 flags of IRQ0ICR register. If IRQ0IR flag has already been set, make sure to clear it.
(5) Set the interrupt operation in STANDBY mode STB_MSK(0x03F5F) bp0: STB_MSK_IRQ0=1	(5) Set the STB_MSK_IRQ0 flag of STB_MSK register so that an interrupt is generated after the transition to STANDBY mode.
(6) Enable interrupts IRQ0ICR(0x03FE2) bp1: IRQ0IE=1	(6) Set the IRQ0IE flag of IRQ0ICR register to “1” to enable interrupts.
(7) Set STOP mode CPUM(0x03F00) bp3: STOP=1	(7) Set the STOP flag of CPUM register to “1” to transit to STOP mode. [Chapter 3 3.3.3 STANDBY Mode]

If the "Low" level signal is input to the external interrupt 0 pin, then, the value of the external interrupt valid edge specification flag (REDG0) and the level of external interrupt 0 pin are matched, the external interrupt 0 is accepted and LSI returns from STOP mode.

The LSI returns from STOP mode after oscillation stabilization wait time set by DLYCTR register has passed since the acceptance of an external interrupt.
[Chapter 2 2.6.2 Oscillation Stabilization Wait time]

To change the value of IRQCNT register, NFnCTR register and LVLMD register, clear STB_EDG register and STB_MSK register to "00".

Settings "Edge interrupt" by the STB_EDG_IRQn flag of STB_EDG register and "Level" by the LVLENn flag of LVLMD register cannot be used at the same time. If using them together, unintended interrupt may be generated.

■ External Interrupt Operation in STANDBY Mode (External Interrupt 0 to 4/Key Interrupt) (Control of Interrupt which is return factor before the transition to STANDBY mode)

This setup example shows the operation to prevent that an interrupt to be the return factor from STANDBY mode is input after the transition to STANDBY mode, and the LSI cannot return from STANDBY mode.

The LSI can return from STANDBY mode by external interrupts. When the interrupt selection is set by STB_MSK register after the transition to STANDBY mode, the external interrupt factor generated before the transition to STANDBY mode can be generated after the transition to STANDBY mode. In this case, an interrupt is accepted after the transition to STANDBY mode and the LSI returns from STANDBY mode immediately.

■ Return by External Interrupt in STANDBY Mode Setup Example 3

This setup example is used to prevent the following cases;
LSI cannot return from STANDBY mode even if an interrupt to be the return factor from STANDBY mode is input just before the transition to STANDBY mode.

An example setup procedure, with a description of each step is shown below.

Setup Procedure	Description
(1) Specify interrupt valid edge IRQ0ICR(0x03FE2) bp5: REDG0=0	(1) Set the REDG0 of IRQ0ICR register to “0” to specify falling edge as the interrupt active edge.
(2) Set the external interrupt pin Pull up the external interrupt 0 pin in advance.	(2) The value of the REDG0 flag of IRQ0ICR register differs from the level of pin IRQ0.
(3) Set the external interrupt IRQCNT(0x03FD0) bp0: P20EN=1	(3) Set the P20EN flag of IRQCNT register to “1” to set P20 as the external interrupt.
(4) Set the interrupt level IRQ0ICR(0x03FE2) bp7 to 6: IRQ0LV1 to 0=10	(4) Set the interrupt level by the IRQ0LV1 to 0 flags of IRQ0ICR register. If IRQ0IR flag has already been set, make sure to clear it.
(5) Set the interrupt operation in STANDBY mode STB_MSK(0x03F5F) bp0: STB_MSK_IRQ0=1	(5) Set the STB_MSK_IRQ0 flag of STB_MSK register so that an interrupt is generated after the transition to STANDBY mode.
(6) Enable interrupts IRQ0ICR(0x03FE2) bp1: IRQ0IE=1	(6) Set the IRQ0IE flag of the IRQ0ICR register to “1” to enable interrupts.
(7) Set STOP mode CPUM(0x03F00) bp3: STOP=1	(7) Set the STOP flag of CPUM register to “1” to transit to STOP mode. [Chapter 3 3.3.3 STANDBY Mode]

If the "Low" level signal is input to the external interrupt 0 pin, then, the value of the external interrupt valid edge specification flag (REDG0) and the level of external interrupt 0 pin are matched, the external interrupt 0 is accepted and LSI returns from STOP mode.

The LSI returns from STOP mode after oscillation stabilization wait time set by DLYCTR register has passed since the acceptance of an external interrupt.
[Chapter 2 2.6.2 Oscillation Stabilization Wait time]

To use these settings indicated in this section, interrupts in STANDBY mode are edge interrupts regardless of STB_EDG register.

To change the value of IRQCNT register, NFnCTR register and LVLMD register, clear STB_EDG register and STB_MSK register to "00".

Settings "Masking operation" by the STB_MSK_IRQn flag of STB_MSK register and "Level" by the LVLENn flag of LVLMD register cannot be used at the same time. If using them together, unintended interrupt may be generated.

When the STB_MSK_IRQn flag of STB_MSK register is set and the LSI returns from STANDBY mode by the external interrupt IRQn generation (including retention request) after the transition to STANDBY mode, the STB_MSK_IRQn flag is automatically turned to “0”.

When the STB_MSK_IRQn flag of STB_MSK register is set and the LSI returns from STANDBY mode by the external interrupt IRQn generation (including external interrupts except IRQn), the STB_MSK_IRQn flag remains set but the mask operation is disabled. It is necessary to clear all the flags of STB_MSK register after return from STANDBY mode.

4.3.9 Noise Filter

■ Noise Filter (External Interrupt 0 to 4)

Noise filter reduces noise by sampling the input waveform from the external interrupt pins (IRQ0 to 4). There are 7 types of sampling cycle (fpll, fpll/2^5, fpll/2^6, fpll/2^7, fpll/2^8, fpll/2^9, fpll/2^{10}, fx).

■ Noise Filter Selection (External Interrupt 0 to 4)

Noise filter can be selected by setting the NFnEN1 flag of NFnCTR register to "1".

Table:4.3.4 Addition of Noise Removal Function

NFnEN1	IRQn input
0	IRQn noise filter OFF
1	IRQn noise filter ON

■ Sampling Cycle Setup (External Interrupt 0 to 4)

The sampling cycle of noise filter can be set by the NFnSCK2 to 0 flags of NFnCTR register.

Table:4.3.5 Sampling Cycle / Time of Noise Filter

NFnSCK2 to 0	Sampling cycle	fpll=10 MHz	
000	fpll	10 MHz	100 ns
001	fpll/2^5	312.5 kHz	3.2 µs
010	fpll/2^6	156.25 kHz	6.4 µs
011	fpll/2^7	78.12 kHz	12.8 µs
100	fpll/2^8	39.06 kHz	25.6 µs
101	fpll/2^9	19.53 kHz	51.20 µs
110	fpll/2^{10}	9.76 kHz	102.40 µs
111	fx	32 kHz	31.25 µs

■ Noise Removal Function Operation (External Interrupts 0 to 4)

After sampling the input signal to the external interrupt pins (IRQ0 to 4) with the set sampling time, if the same level is received for three times continuously, that level is sent to the LSI. If the same level is not received for three times continuously, the previous level is sent.

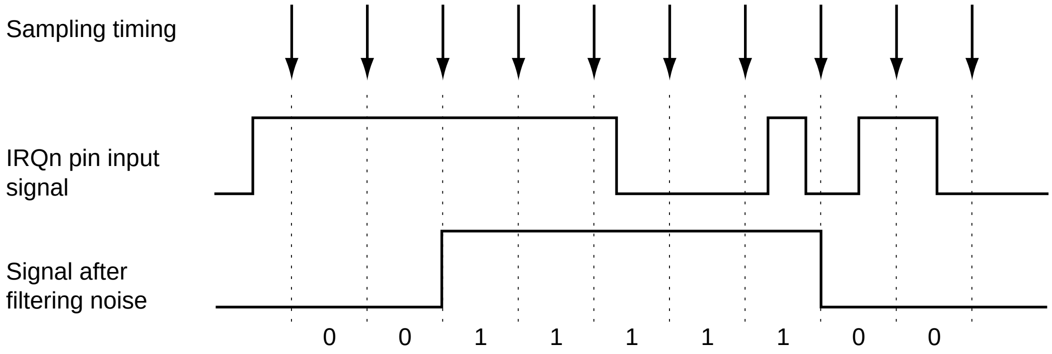


Figure:4.3.10 Operation of the Noise Filter

Noise filter cannot be used in STOP mode and HALT mode.
Set the NFnEN1 flag to "0" when using IRQ0 to 4 for returning from STANDBY mode.

■ Noise Filter Setup Example (External Interrupt 0)

Noise removal function is added to the input signal from P20 pin to generate the external interrupt 0 (IRQ0) at the rising edge. The sampling clock is set to fpll, and the operation state is fpll = 10 MHz. The following shows an example of setup procedures.

Setup Procedure	Description
(1) External interrupt setup IRQCNT (0x03F3D0) bp0: P20EN =1	(1) Set the P20EN flag of IRQCNT register to "1" to set P20 to the external interrupt.
(2) Specify the interrupt valid edge IRQ0ICR (0x03FE2) bp5: REDG0 =1	(2) Set the REDG0 flag of IRQ0ICR register to "1" to specify the interrupt valid edge to the rising edge.
(3) Select the sampling clock NF0CTR(0x03FD1) bp7 to 5: NF0SCK2 to 0=000	(3) Select the sampling clock to fpll by the NF0SCK2 to 0 flags of NF0CTR register.
(4) Set the noise filter operation NF0CTR (0x03FD1) bp4: NF0EN1 =1	(4) Set the NF0EN1 flag of NF0CTR register to "1" to add the noise filter operation.
(5) Set the interrupt level IRQ0ICR (0x03FE2) bp7 to 6: IRQ0LV1 to 0 =10	(5) Set the interrupt level by setting the IRQ0LV1 to 0 flags of IRQ0ICR register. If IRQ0IR flag has already been set, make sure to clear it. [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup]
(6) Enable the interrupt IRQ0ICR (0x03FE2) bp1: IRQ0IE =1	(6) Set the IRQ0IE flag of IRQ0ICR register to "1" to enable the interrupt.

*Above (3) and (4) can be set at the same time.

The input signal from P20 pin outputs the interrupt factor at the edge based on the programmable active edge after passing through the noise filter.

The noise filter should be set before interrupts are enabled.

External interrupt pins need to be pulled up in advance to prevent intermediate potential input.

Chapter 5 I/O Port

5.1 Overview

5.1.1 I/O Port Overview

This LSI has a total of 70 pins, including the shared special function pins which are allocated to the I/O ports (Port 0, Port 2, Port 3, Port 4, Port 5, Port 6, Port 7, Port 8, Port 9, Port A and Port B).

5.1.2 I/O Port Status at Reset

Table:5.1.1 I/O port status at reset (single chip mode)

Port	I/O mode	Pull-up/pull-down resistor	I/O port/ Special functions
Port 0	Input mode	No pull-up resistor	I/O port
Port 2	Input mode	P27: Pull-up resistor	I/O port
		Others: No pull-up resistor	
Port 3	Input mode	No pull-up/pull-down resistor	I/O port
Port 4	Input mode	No pull-up resistor	I/O port
Port 5	Input mode	No pull-up resistor	I/O port
Port 6	Input mode	No pull-up resistor	I/O port
Port 7	Input mode	No pull-up resistor	I/O port
Port 8	Input mode	No pull-up resistor	I/O port
Port 9	Input mode	No pull-up/pull-down resistor	I/O port
Port A	Input mode	No pull-up resistor	I/O port
Port B	Input mode	No pull-up/pull-down resistor	I/O port

The values of pull-up/pull-down resistors should be calculated in the following ways based on the electrical characteristics in LSI User's Manual of each model.

How to determine a pull-up resistor value
ex) When pins maintain at the low-level value which guarantee the performance based on the electrical characteristics, V_{DD5}=5 V, V_{IN}=V_{SS} and input current is Min=-50 µA, Typ=-100 µA, Max=-500 µA (minus (-) refer to the current passing from microcontroller).
The resistor value of Typ=50 Ω can converted based on the values above.
However, the value may vary significantly depending on the temperature.
Note that this value varies widely depending on the temperature.
At temperature variation from -40 °C to 85 °C,
the resistor values vary from Min=10 kΩ to Max=100 kΩ.

How to determine a pull-down resistor value
ex) When pins maintain at the high level value which guarantee the performance based on the electrical characteristics, V_{DD5}=5 V, V_{IN}=V_{DD5} and input current is Min=50 µA, Typ=100 µA, Max=500 µA.
The resistor value of Typ=50 Ω can converted based on the values above.
However, the value may vary significantly depending on the temperature.
At temperature variation from -40 °C to 85 °C,
the resistor value vary from Min=10 kΩ to Max=100 kΩ.

5.2 Control Registers

Each port is controlled by the data output register (PnOUT), the data input register (PnIN), the I/O direction control register (PnDIR), the pull-up resistor control register (PnPLU) or the pull-up/pull-down resistor control register (SELUD, SELUD2, PnPLUD) and registers that control special function pin (PnOMD, PnOMD1, PnOMD2, PnIMD, PnODC, LEDCNT). The following table shows the list of registers.

Table:5.2.1 I/O Port Control Registers List

Table remarks √: With function -: Without function

Register	Address	R/W	Function	Page	MN101EFA8/A3	MN101EFA7/A2
P0OUT	0x03E70	R/W	Port 0 output register	V-9	√	√
P0IN	0x03E80	R	Port 0 input register	V-9	√	√
P0DIR	0x03E90	R/W	Port 0 direction control register	V-10	√	√
P0PLU	0x03EA0	R/W	Port 0 pull-up resistor control register	V-11	√	√
P0OMD1	0x03EB0	R/W	Port 0 output mode register 1	V-12	√	√
P0OMD2	0x03EC0	R/W	Port 0 output mode register 2	V-12	√	√
P0ODC	0x03EF0	R/W	Port 0 Nch open-drain control register	V-13	√	√
P2OUT	0x03E72	R/W	Port 2 output register	V-19	√	√
P2IN	0x03E82	R	Port 2 input register	V-19	√	√
P2DIR	0x03E92	R/W	Port 2 direction control register	V-19	√	√
P2PLU	0x03EA2	R/W	Port 2 pull-up resistor control register	V-20	√	√
P3OUT	0x03E73	R/W	Port 3 output register	V-26	√	-
P3IN	0x03E83	R	Port 3 input register P3PLUD	V-26	√	-
P3DIR	0x03E93	R/W	Port 3 direction control register	V-27	√	-
P3PLUD	0x03EA3	R/W	Port 3 pull-up/pull-down resistor control register	V-27	√	-
P3IMD	0x03EC3	R/W	Port 3 input mode register	V-28	√	-
P3ODC	0x03EF3	R/W	Port 3 Nch open-drain control register	V-28	√	-
SELUD	0x03EAF	R/W	Pull-up/pull-down resistor selection register	V-29	√	√
P4OUT	0x03E74	R/W	Port 4 output register	V-38	√	-
P4IN	0x03E84	R	Port 4 input register	V-38	√	-
P4DIR	0x03E94	R/W	Port 4 direction control register	V-39	√	-
P4PLU	0x03EA4	R/W	Port 4 pull-up resistor control register	V-39	√	-
P4ODC	0x03EF4	R/W	Port 4 Nch open-drain control register	V-40	√	-

R/W: Readable/Writable

Table remarks √: With function -: Without function

Register	Address	R/W	Function	Page	MN101EFA8/A3	MN101EFA7/A2
P5OUT	0x03E75	R/W	Port 5 output register	V-46	√	√
P5IN	0x03E85	R	Port 5 input register	V-46	√	√
P5DIR	0x03E95	R/W	Port 5 direction control register	V-46	√	√
P5PLU	0x03EA5	R/W	Port 5 pull-up resistor control register	V-47	√	√
P5OMD	0x03EB5	R/W	Port 5 output mode register	V-47	√	√
P5IMD	0x03EC5	R/W	Port 5 input mode register	V-48	-	√
P5ODC	0x03EF5	R/W	Port 5 Nch open-drain control register	V-48	√	√
P6OUT	0x03E76	R/W	Port 6 output register	V-70	√	√
P6IN	0x03E86	R	Port 6 input register	V-70	√	√
P6DIR	0x03E96	R/W	Port 6 direction control register	V-70	√	√
P6PLU	0x03EA6	R/W	Port 6 pull-up resistor control register	V-71	√	√
P6OMD	0x03EB6	R/W	Port 6 output mode register	V-71	√	√
P6ODC	0x03EF6	R/W	Port 6 Nch open-drain control register	V-72	√	√
P7OUT	0x03E77	R/W	Port 7 output register	V-79	√	√
P7IN	0x03E87	R	Port 7 input register	V-79	√	√
P7DIR	0x03E97	R/W	Port 7 direction control register	V-79	√	√
P7PLU	0x03EA7	R/W	Port 7 pull-up resistor control register	V-80	√	√
P7ODC	0x03EF7	R/W	Port 7 Nch open-drain control register	V-80	√	√
P8OUT	0x03E78	R/W	Port 8 output register	V-86	√	√
P8IN	0x03E88	R	Port 8 input register	V-86	√	√
P8DIR	0x03E98	R/W	Port 8 direction control register	V-86	√	√
P8PLU	0x03EA8	R/W	Port 8 pull-up resistor control register	V-87	√	√
P8OMD	0x03EB8	R/W	Port 8 output mode register	V-87	√	√
P9OUT	0x03E79	R/W	Port 9 output register	V-92	√	√
P9IN	0x03E89	R	Port 9 input register	V-93	√	√
P9DIR	0x03E99	R/W	Port 9 direction control register	V-94	√	√
P9PLUD	0x03EA9	R/W	Port 9 pull-up/pull-down resistor control register	V-95	√	√
P9IMD	0x03EC9	R/W	Port 9 input mode register	V-96	√	√

R/W: Readable/Writable

Table remarks √: With function -: Without function

Register	Address	R/W	Function	Page	MN101EFA8/A3	MN101EFA7/A2
SELUD2	0x03EBF	R/W	Pull-up/pull-down resistor selection register 2	V-97	√	√
PAOUT	0x03E7A	R/W	Port A output register	V-108	√	√
PAIN	0x03E8A	R	Port A input register	V-108	√	√
PADIR	0x03E9A	R/W	Port A direction control register	V-108	√	√
PAPLU	0x03EAA	R/W	Port A pull-up resistor control register	V-109	√	√
PAOMD	0x03EBA	R/W	Port A output mode register	V-109	√	√
PAIMD	0x03ECA	R/W	Port A input mode register	V-110	√	√
LEDCNT	0x03EE0	R/W	Port LED control register	V-111	√	√
PBOUT	0x03E7B	R/W	Port B output register	V-117	√	-
PBIN	0x03E8B	R	Port B input register	V-117	√	-
PBDIR	0x03E9B	R/W	Port B direction control register	V-117	√	-
PBPLUD	0x03EAB	R/W	Port B pull-up/pull-down resistor control register	V-118	√	-
PBIMD	0x03ECB	R/W	Port B input mode register	V-118	√	-

R/W: Readable/Writable

5.3 Port 0

5.3.1 Description

MN101EFA7/A2 do not include P07.

■ General Port Setup

To output data to pin, set the control flag of P0DIR register to "1" and write the value of P0OUT register.

To read input data of pin, set the control flag of P0DIR register to "0" and read the value of P0IN register.

Each bit can be set individually as either an input or output by P0DIR register. The control flag of P0DIR register is set to "1" for output mode, and "0" for input mode.

Each bit can be set individually if pull-up resistor is added or not, by P0PLU register. Set the control flag of P0PLU register to "1" to add pull-up resistor.

For P03 and P04 each bit can be selected individually as Nch open-drain output by P0ODC register. P0ODC register is set to "1" for Nch open-drain output, and "0" for push-pull output.

■ Special Function Pin Setup

P01 is also used as the I/O pin of timer 9.

P02 is also used as the I/O pin of timer 7.

P03 is also used as the I/O pin of timer 8.

P04 is also used as the I/O pin of timer 0 and timer 2.

Each bit for the I/O mode can be set individually by P0OMD1 and P0OMD2 registers. These registers are set to "1" to output the special function data, and "0" to be used as the general port.

P02 is also used as the input pin of serial 0 reception data and UARD0 reception data. When the SC0MD1 flag of SC0MD1 register is "1", P02 is the serial data input pin.

P03 is also used as the I/O pin of serial 0 transmission/reception data and UART0 transmission/reception data. When the SC0SBOS flag of SC0MD1 register is "1", P03 is the serial data I/O pin. Push-pull output or Nch open-drain output can be selected by setting P0ODC register.

P04 is also used as the I/O pin of serial 0 clock. When the SC0SBTS flag of the serial clock. Push-pull output or Nch open-drain output can be selected by setting P0ODC register.

Table:5.3.1 The Port 0 Special Function Pins

Table remarks √: With function -: Without function

Pins	Special Functions	MN101EFA8	MN101EFA3	MN101EFA7	MN101EFA2
P00	OCD_DATA	√	√	√	√
P01	TM9IOB	√	√	√	√
	OCD_CLK	√	√	√	√
P02	TM7IOB	√	√	√	√
	SBI0A	√	√	√	√
	RXD0A	√	√	√	√
P03	TM8IOB	√	√	√	√
	SBO0A	√	√	√	√
	TXD0A	√	√	√	√
P04	TM0IOB	√	√	√	√
	TM2IOB	√	√	√	√
	SBT0A	√	√	√	√
P05	-	√	√	√	√
P06	-	√	√	√	√
P07	-	√	√	-	-

5.3.2 Registers

- Port 0 Output Register (P0OUT: 0x03E70)

MN101EFA8/MN101EFA3

bp	7	6	5	4	3	2	1	0
Flag	P0OUT7	P0OUT6	P0OUT5	P0OUT4	P0OUT3	P0OUT2	P0OUT1	P0OUT0
At reset	X	X	X	X	X	X	X	X
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-0	P0OUT7-0	Output data 0: Output "Low" (V_{SS} level) 1: Output "High" (V_{DD5} level)

- Port 0 Output Register (P0OUT: 0x03E70)

MN101EFA7/MN101EFA2

bp	7	6	5	4	3	2	1	0
Flag	-	P0OUT6	P0OUT5	P0OUT4	P0OUT3	P0OUT2	P0OUT1	P0OUT0
At reset	-	X	X	X	X	X	X	X
Access	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7	-	-
6-0	P0OUT6-0	Output data 0: Output "Low" (V_{SS} level) 1: Output "High" (V_{DD5} level)

- Port 0 Input Register (P0IN: 0x03E80)

MN101EFA8/MN101EFA3

bp	7	6	5	4	3	2	1	0
Flag	P0IN7	P0IN6	P0IN5	P0IN4	P0IN3	P0IN2	P0IN1	P0IN0
At reset	X	X	X	X	X	X	X	X
Access	R	R	R	R	R	R	R	R

bp	Flag	Description
7-0	P0IN7-0	Input data 0: Pin is "Low" (V_{SS} level) 1: Pin is "High" (V_{DD5} level)

■ Port 0 Input Register (P0IN: 0x03E80)

MN101EFA7/MN101EFA2

bp	7	6	5	4	3	2	1	0
Flag	-	P0IN6	P0IN5	P0IN4	P0IN3	P0IN2	P0IN1	P0IN0
At reset	-	X	X	X	X	X	X	X
Access	-	R	R	R	R	R	R	R

bp	Flag	Description
7	-	-
6-0	P0IN6-0	Input data 0: Pin is "Low" (V_{SS} level) 1: Pin is "High" (V_{DD5} level)

■ Port 0 Direction Control Register (P0DIR: 0x03E90)

MN101EFA8/MN101EFA3

bp	7	6	5	4	3	2	1	0
Flag	P0DIR7	P0DIR6	P0DIR5	P0DIR4	P0DIR3	P0DIR2	P0DIR1	P0DIR0
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-0	P0DIR7-0	I/O mode selection 0: Input mode 1: Output mode

■ Port 0 Direction Control Register (P0DIR: 0x03E90)

MN101EFA7/MN101EFA2

bp	7	6	5	4	3	2	1	0
Flag	-	P0DIR6	P0DIR5	P0DIR4	P0DIR3	P0DIR2	P0DIR1	P0DIR0
At reset	-	0	0	0	0	0	0	0
Access	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7	-	-
6-0	P0DIR6-0	I/O mode selection 0: Input mode 1: Output mode

■ Port 0 Pull-up Resistor Control Register (P0PLU: 0x03EA0)

MN101EFA8/MN101EFA3

bp	7	6	5	4	3	2	1	0
Flag	P0PLU7	P0PLU6	P0PLU5	P0PLU4	P0PLU3	P0PLU2	P0PLU1	P0PLU0
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-0	P0PLU7-0	Pull-up resistor selection 0: Not added 1: Added

■ Port 0 Pull-up Resistor Control Register (P0PLU: 0x03EA0)

MN101EFA7/MN101EFA2

bp	7	6	5	4	3	2	1	0
Flag	-	P0PLU6	P0PLU5	P0PLU4	P0PLU3	P0PLU2	P0PLU1	P0PLU0
At reset	-	0	0	0	0	0	0	0
Access	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7	-	-
6-0	P0PLU6-0	Pull-up resistor selection 0: Not added 1: Added

- Port 0 Output Mode Register 1 (P0OMD1: 0x03EB0)

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	P0OMD14	P0OMD13	P0OMD12	P0OMD11	-
At reset	-	-	-	0	0	0	0	-
Access	-	-	-	R/W	R/W	R/W	R/W	-

bp	Flag	Description
7-5	-	-
4	P0OMD14	I/O port or special function selection 0: P04 1: TM0IOB/TM2IOB
3	P0OMD13	I/O port or special function selection 0: P03 1: TM8IOB
2	P0OMD12	I/O port or special function selection 0: P02 1: TM7IOB
1	P0OMD11	I/O port or special function selection 0: P01 1: TM9IOB
0	-	-

- Port 0 Output Mode Register 2 (P0OMD2: 0x03EC0)

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	P0OMD24	-	-	-	-
At reset	-	-	-	0	-	-	-	-
Access	-	-	-	R/W	-	-	-	-

bp	Flag	Description
7-5	-	-
4	P0OMD24	P04 special function selection 0: TM0IOB 1: TM2IOB
3-0	-	-

■ Port 0 Nch Open-drain Control Register (P0ODC: 0x03EF0)

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	P0ODC4	P0ODC3	-	-	-
At reset	-	-	-	0	0	-	-	-
Access	-	-	-	R/W	R/W	-	-	-

bp	Flag	Description
7-5	-	-
4	P0ODC4	P04 Nch open-drain output selection 0: Push-pull output 1: Nch open-drain output
3	P0ODC3	P03 Nch open-drain output selection 0: Push-pull output 1: Nch open-drain output
2-0	-	-

5.3.3 Block Diagram

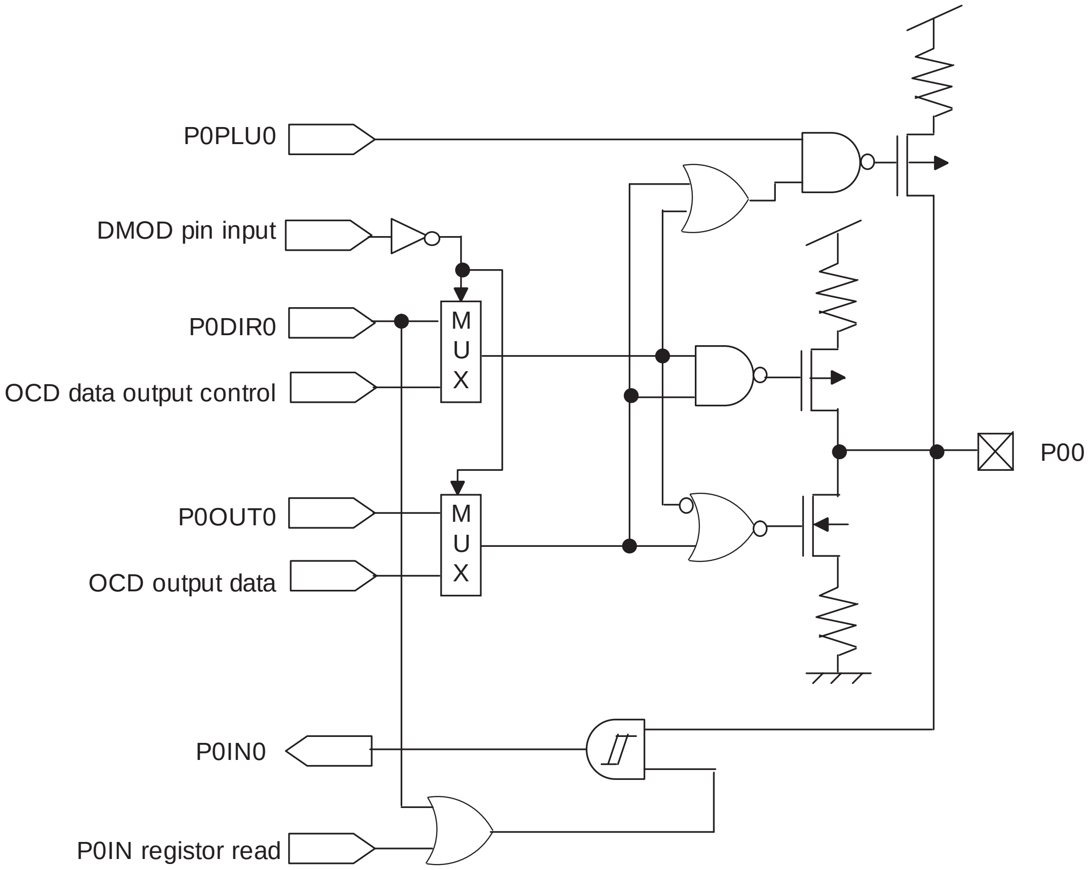


Figure:5.3.1 P00 Block Diagram (MN101EFA8/A7/A3/A2)

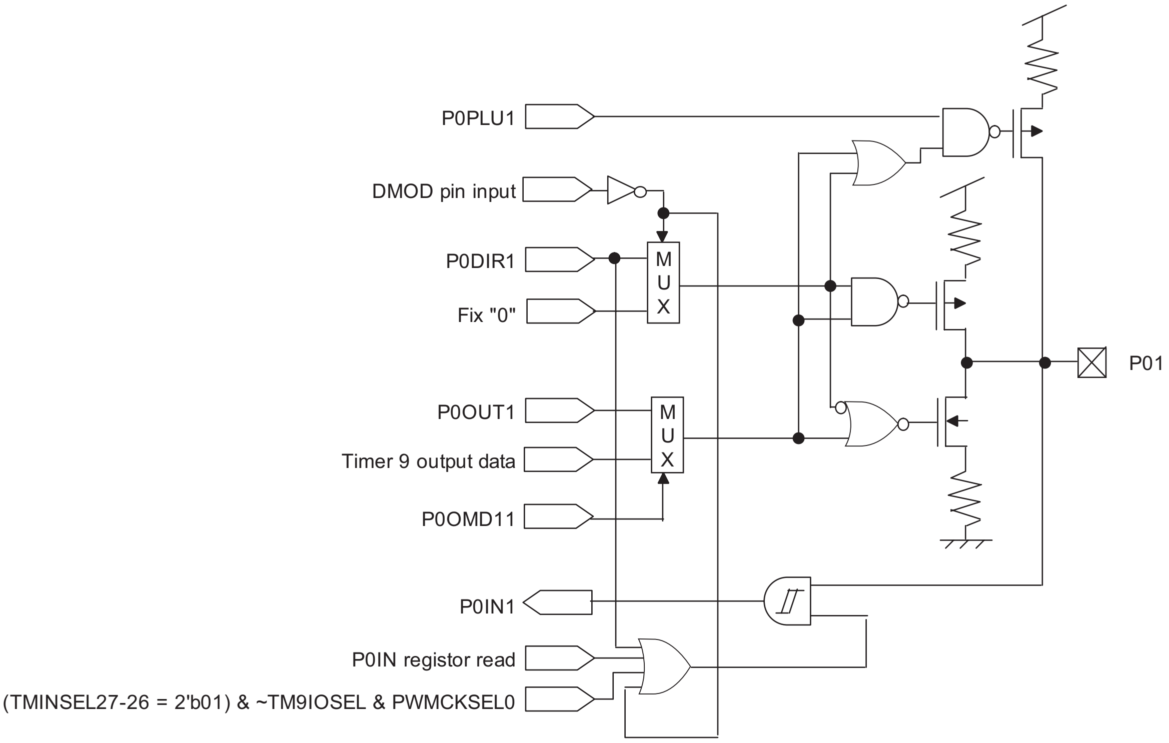


Figure:5.3.2 P01 Block Diagram (MN101EFA8/A7/A3/A2)

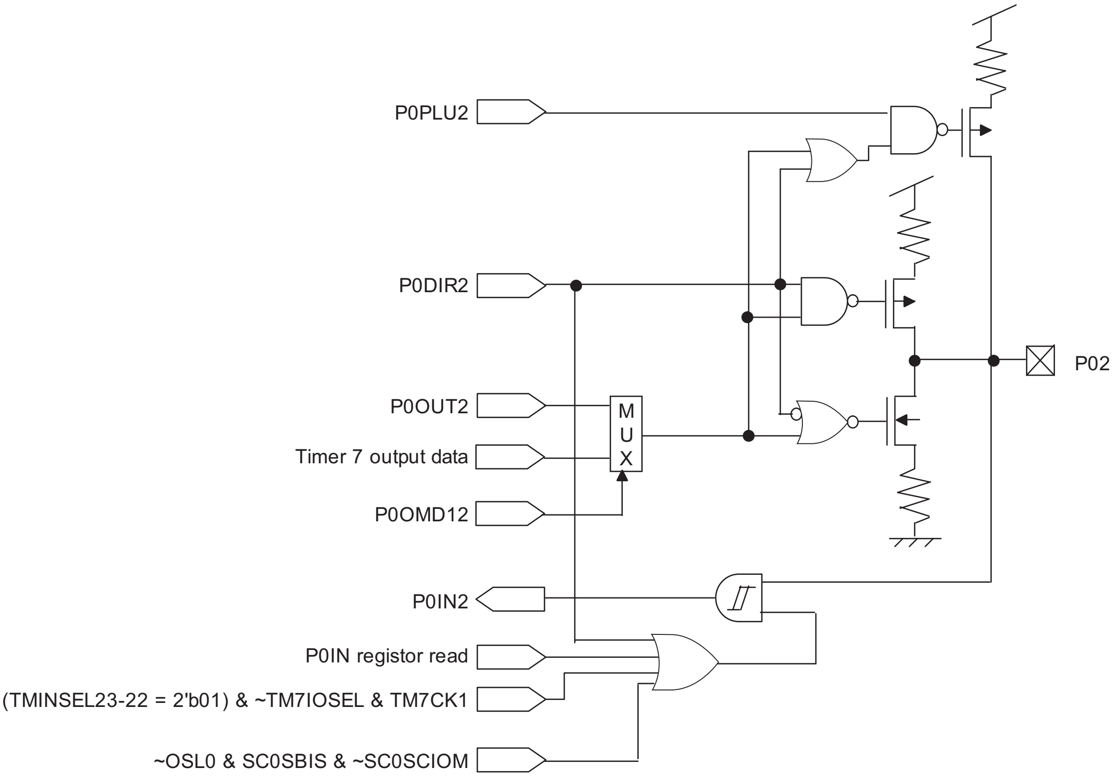


Figure:5.3.3 P02 Block Diagram (MN101EFA8/A7/A3/A2)

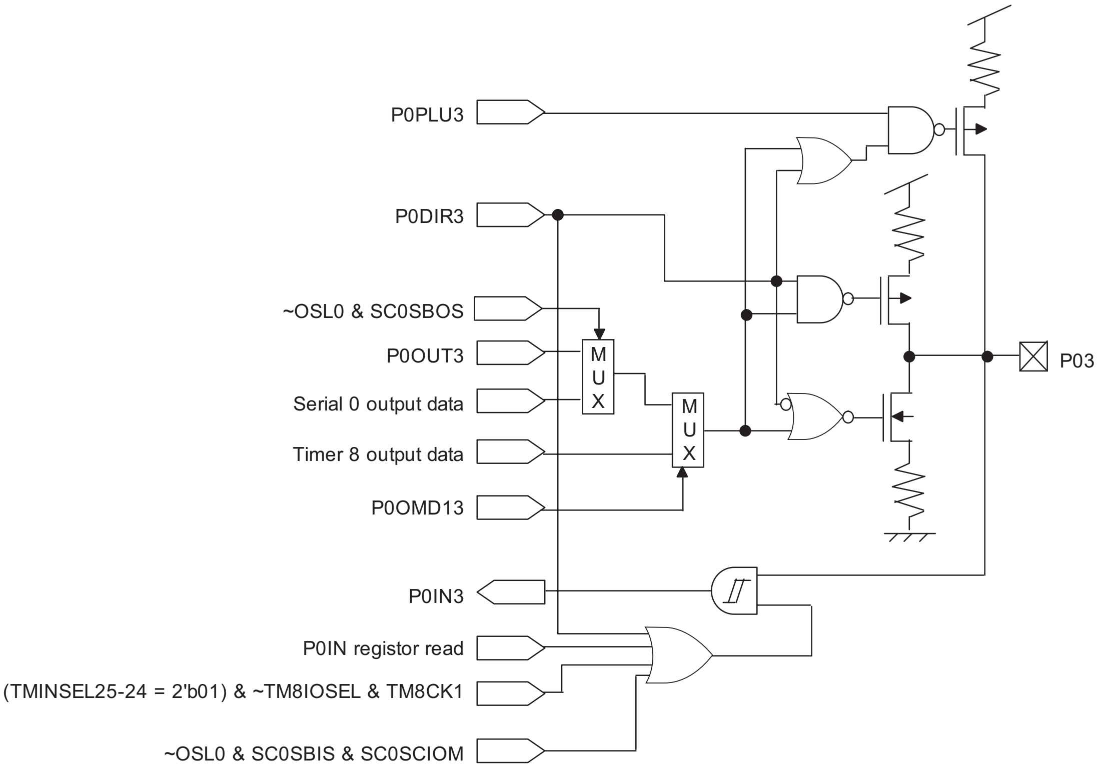


Figure:5.3.4 P03 Block Diagram (MN101EFA8/A7/A3/A2)

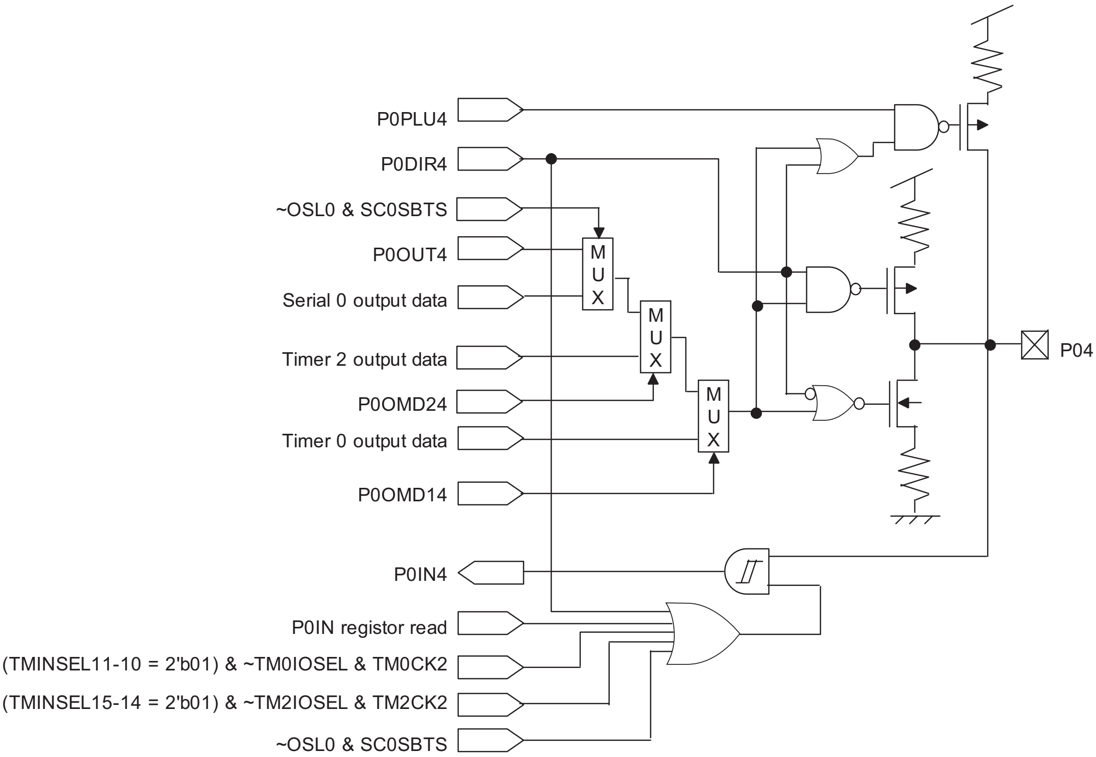
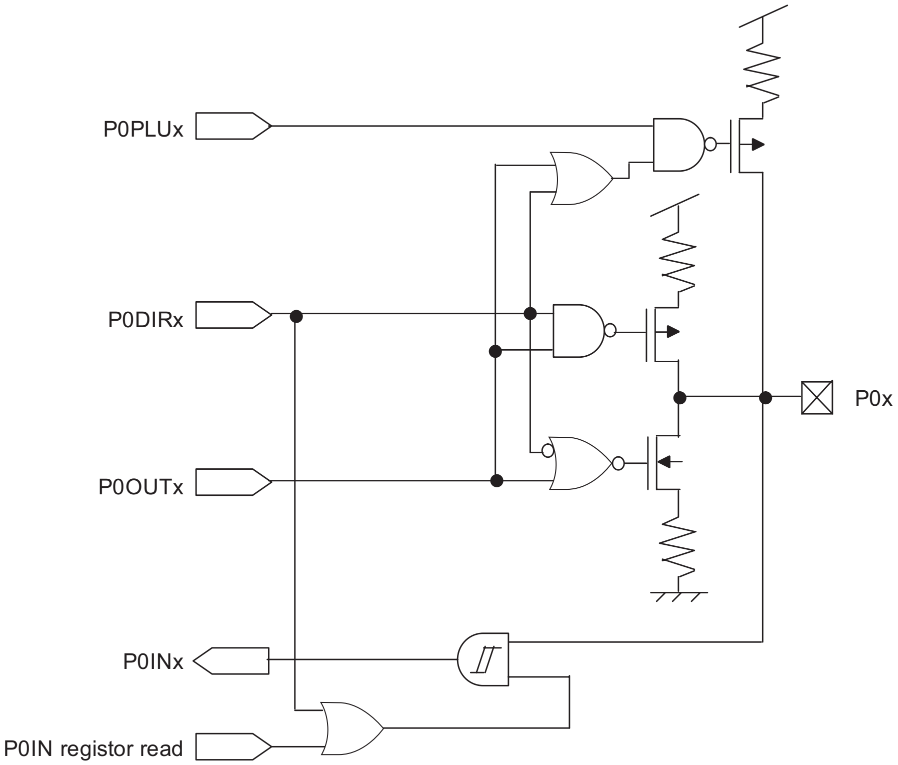


Figure:5.3.5 P04 Block Diagram (MN101EFA8/A7/A3/A2)



* x is 5 to 7
* P05,P06 is mounted MN101EFA8/A7/A3/A2
* P07 is mounted MN101EFA8/A3

Figure:5.3.6 P0x Block Diagram (MN101EFA8/A7/A3/A2)

5.4 Port 2

5.4.1 Description

■ General Port Setup

To output data to pins, set the control flag of P2DIR register to "1" and write the data to P2OUT register.

To read input data of pins, set the control flag of P2DIR register to "0" and read the value of P2IN register.

Each bit can be set individually as either an input or output by P2DIR register.
The control flag of P2DIR register is set to "1" for output mode, and "0" for input mode.

Each bit can be set individually if pull-up resistor is added or not, by P2PLU register.
Set the control flag of P2PLU register to "1" to add pull-up resistor.

P27 is the reset pin.
To execute soft reset, write "0" to the P2OUT7 flag of P2OUT register. Pull-up resistor is always added to P27.

■ Special Function Pin Setup

P20 to P24 are also used as external interrupt pins.

P25 is also used as the input pin of external high speed oscillation.

P26 is also used as the output pin of external high speed oscillation. To use P26 as external oscillation pin, set the HOSCCNT flag of external oscillation control register (OSCCNT) to "1".

Table:5.4.1 The Port 2 Special Function Pins

Table remarks √: With function -: Without function

Pins	Special Functions	MN101EFA8	MN101EFA3	MN101EFA7	MN101EFA2
P20	IRQ0	√	√	√	√
P21	IRQ1	√	√	√	√
P22	IRQ2	√	√	√	√
P23	IRQ3	√	√	√	√
P24	IRQ4	√	√	√	√
P25	OSC1	√	√	√	√
P26	OSC2	√	√	√	√
P27	NRST	√	√	√	√

5.4.2 Registers

- Port 2 Output Register (P2OUT: 0x03E72)

bp	7	6	5	4	3	2	1	0
Flag	P2OUT7	P2OUT6	P2OUT5	P2OUT4	P2OUT3	P2OUT2	P2OUT1	P2OUT0
At reset	1	x	x	x	x	x	x	x
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7	P2OUT7	Output data (reset output) 0: Output "Low" (V_{SS} level) 1: Output "High" (V_{DD5} level)
6-0	P2OUT6-0	Output data 0: Output "Low" (V_{SS} level) 1: Output "High" (V_{DD5} level)

- Port 2 Input Register (P2IN: 0x03E82)

bp	7	6	5	4	3	2	1	0
Flag	P2IN7	P2IN6	P2IN5	P2IN4	P2IN3	P2IN2	P2IN1	P2IN0
At reset	1	x	x	x	x	x	x	x
Access	R	R	R	R	R	R	R	R

bp	Flag	Description
7-0	P2IN7-0	Input data 0: Pin is "Low" (V_{SS} level) 1: Pin is "High" (V_{DD5} level)

- Port 2 Direction Control Register (P2DIR: 0x03E92)

bp	7	6	5	4	3	2	1	0
Flag	-	P2DIR6	P2DIR5	P2DIR4	P2DIR3	P2DIR2	P2DIR1	P2DIR0
At reset	-	0	0	0	0	0	0	0
Access	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7	-	-
6-0	P2DIR6-0	I/O mode selection 0: Input mode 1: Output mode

■ Port 2 Pull-up Resistor Control Register (P2PLU: 0x03EA2)

bp	7	6	5	4	3	2	1	0
Flag	-	P2PLU6	P2PLU5	P2PLU4	P2PLU3	P2PLU2	P2PLU1	P2PLU0
At reset	-	0	0	0	0	0	0	0
Access	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7	-	-
6-0	P2PLU6-0	Pull-up resistor selection 0: Not added 1: Added

5.4.3 Block Diagram

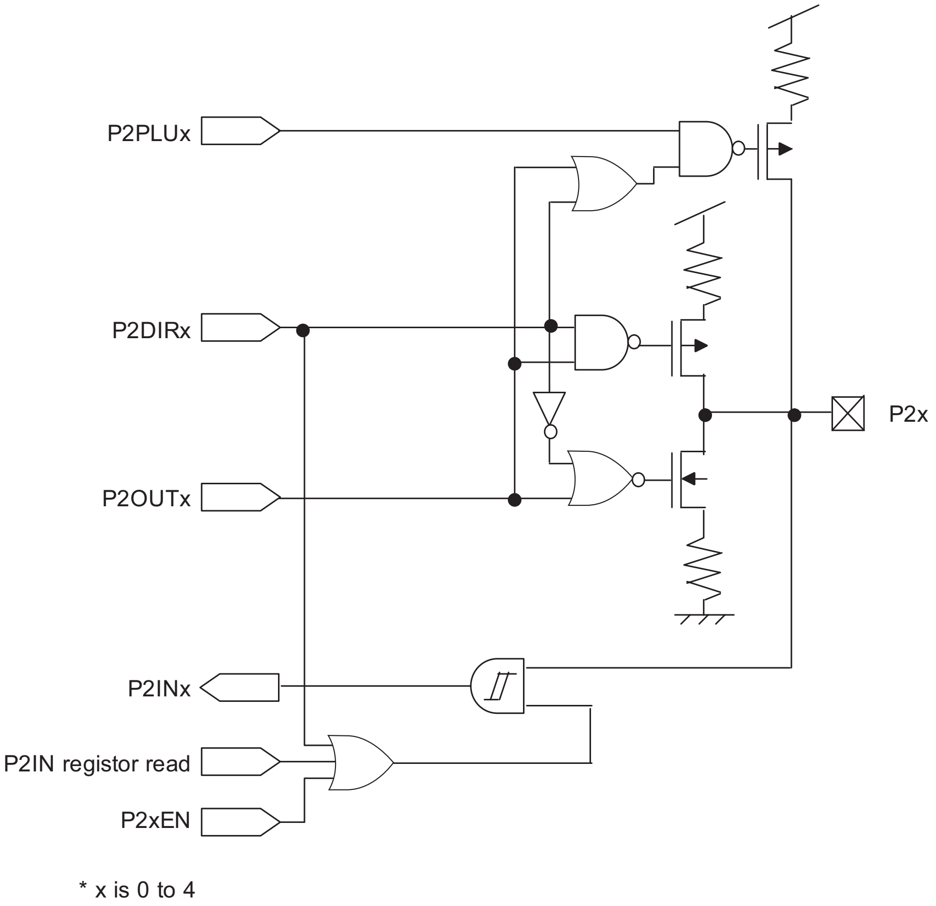


Figure:5.4.1 P2x Block Diagram (MN101EFA8/A7/A3/A2)

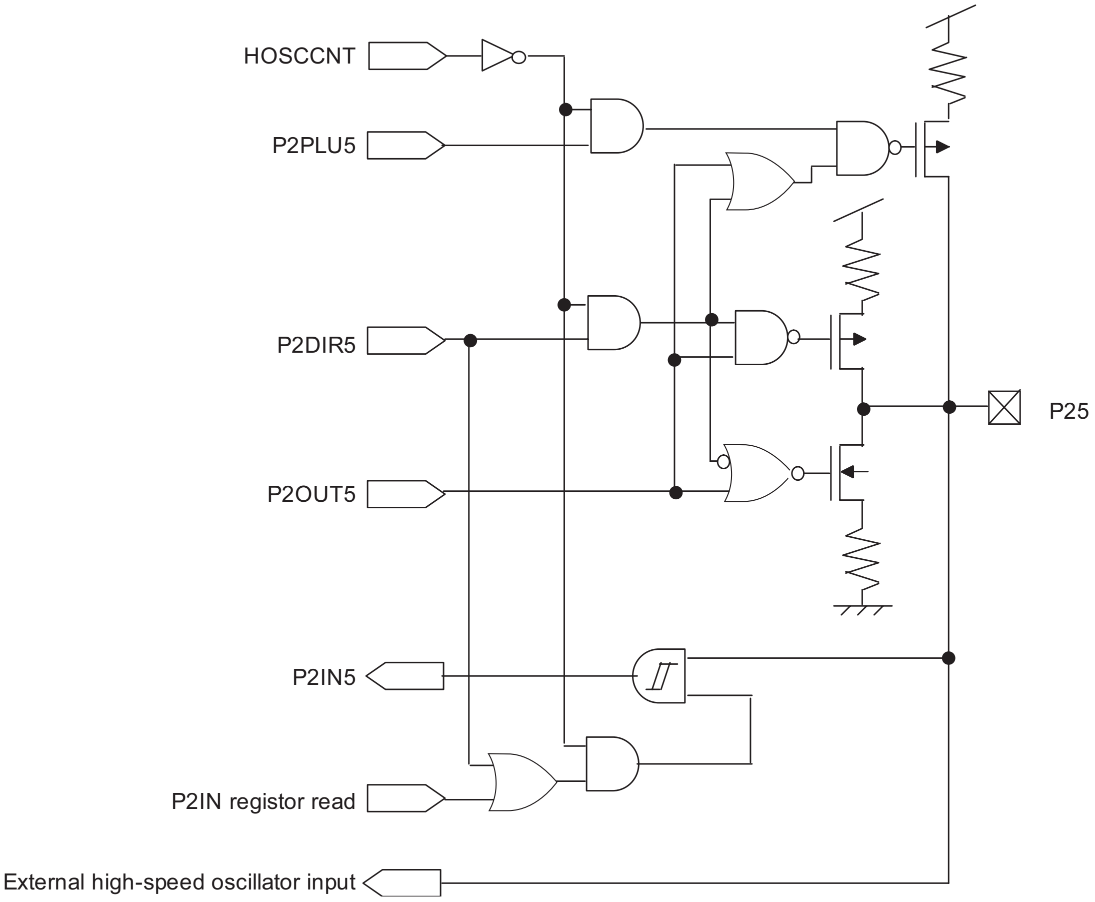


Figure:5.4.2 P25 Block Diagram (MN101EFA8/A7/A3/A2)

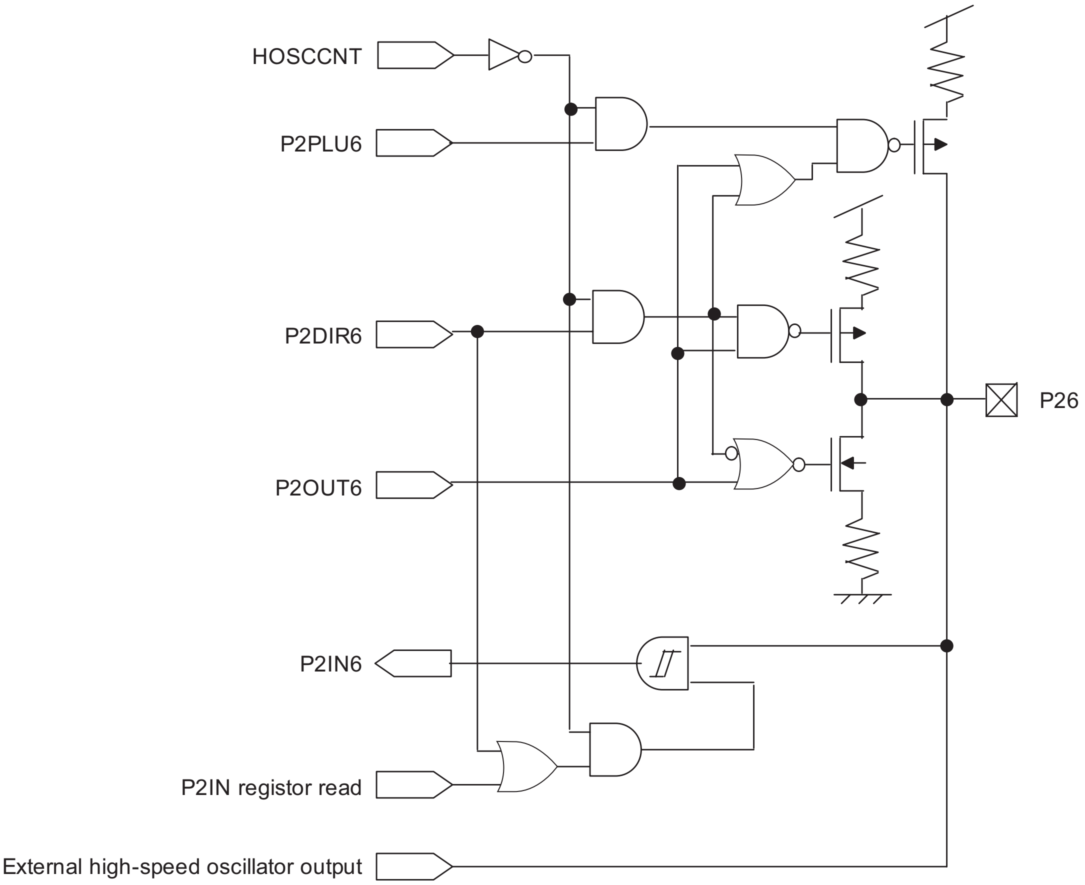


Figure:5.4.3 P26 Block Diagram (MN101EFA8/A7/A3/A2)

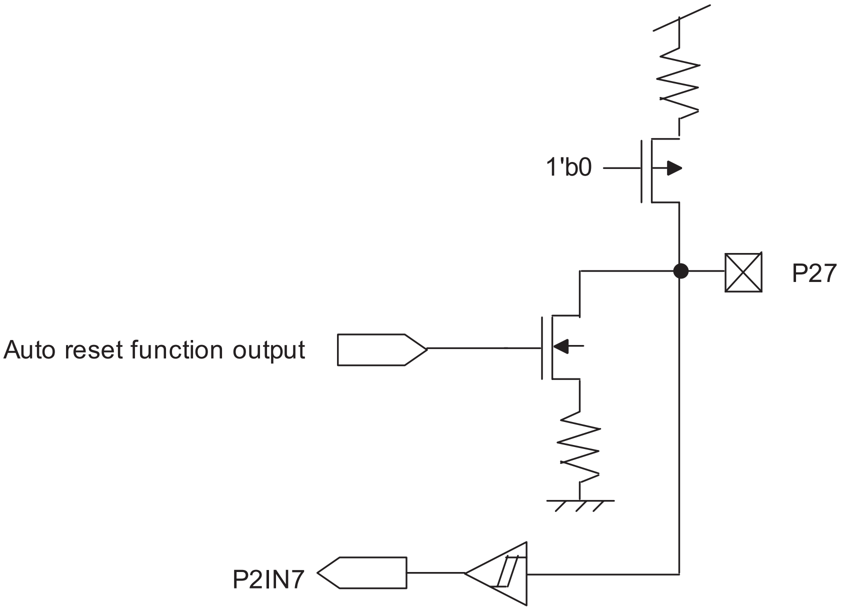


Figure:5.4.4 P27 Block Diagram (MN101EFA8/A7/A3/A2)

5.5 Port 3

5.5.1 Description

MN101EFA7/A2 do not include the Port 3.

■ General Port Setup

To output data to pin, set the control flag of P3DIR register to "1" and write the value of P3OUT register.
To read input data of pin, set the control flag of P3DIR register to "0" and read the value of P3IN register.

Each bit can be set individually as either an input or output by P3DIR register. The control flag of P3DIR register is set to "1" for output mode, and "0" for input mode.

Each bit can be set individually if pull-up resistor is added or not by P3PLUD register. Set the control flag of P3PLUD register to "1" to add pull-up or pull-down resistor.
Port 3 can be selected to add pull-up or pull-down register by the SELUD3 flag of SELUD register.

Each bit can be selected individually as Nch open-drain output by P3ODC register. P3ODC register is set to "1" for Nch open-drain output, and "0" for push-pull output.

Each bit can be selected individually as input mode by P3IMD register. The control flag P3IMD register is set to “1” for input special function data and the value of P3IN register is read to be "1". Also, the flag of P3IMD is set to "0" to be used as the general port.

■ Special Function Pin Setup

P33 is also used as the I/O pin of serial 4 transmission/reception data and IIC4 transmission/reception data.When the SC4SBOS flag of SC4MD1 register is "1", P33 is the I/O pin of the serial data. Push-pull output or Nch open-drain output can be selected by setting P3ODC register.

P34 is also used as the I/O pin of serial 1 transmission/reception data and output pin of URAT1 transmission data. When the SC1SBOS flag of the serial interface 1 mode register 1 (SC1MD1) is "1", P34 is the I/O pin of the serial data. Push-pull output or Nch open-drain output can be selected by setting P3ODC register.

P35 is also used as the input pin of serial 4 reception data. When the SC4SBIS flag of serial 4 mode register is (SC4MD1) is "1", P35 is the input pin of serial data.

P33 is also used as analog input pin. Input mode for each bit can be selected by P3IMD register, When P33 is used as analog input pin, the value read from the port 3 input register is read to be "0".

(This function is equipped in MN101EFA8).
P33 is also used as input pins for touch sensor timer. Set "Used" to corresponding channel by TS0TCHSEL register. Refer to [Chapter XV Touch Sensor Timer]

Table:5.5.1 Port 3 Special Function Pins

Table remarks √: With function -: Without function

Pins	Special Functions	MN101EFA8	MN101EFA3	MN101EFA7	MN101EFA2
P33	SB04B	√	√	-	-
	SDA4B	√	√	-	-
	AN15	√	√	-	-
	TS0IN7	√	-	-	-
P34	SBT4B	√	√	-	-
	SCL4B	√	√	-	-
	TS0RC	√	-	-	-
P35	SBI4B	√	√	-	-
	TS0OP	√	-	-	-

5.5.2 Registers

■ Port 3 Output Register (P3OUT: 0x03E73)

MN101EFA8/MN101EFA3

bp	7	6	5	4	3	2	1	0
Flag	-	-	P3OUT5	P3OUT4	P3OUT3	-	-	-
At reset	-	-	X	X	X	-	-	-
Access	-	-	R/W	R/W	R/W	-	-	-

bp	Flag	Description
7-6	-	-
5-3	P3OUT5-3	Output data 0: Output "Low" (V_{SS} level) 1: Output "High" (V_{DD5} level)
2-0	-	-

■ Port 3 Input Register (P3IN: 0x03E83)

MN101EFA8/MN101EFA3

bp	7	6	5	4	3	2	1	0
Flag	-	-	P3IN5	P3IN4	P3IN3	-	-	-
At reset	-	-	X	X	X	-	-	-
Access	-	-	R	R	R	-	-	-

bp	Flag	Description
7-6	-	-
5-3	P3IN5-3	Input data 0: Pin is "Low" (V_{SS} level) 1: Pin is "High" (V_{DD5} level)
2-0	-	-

■ Port 3 Direction Control Register (P3DIR: 0x03E93)

MN101EFA8/MN101EFA3

bp	7	6	5	4	3	2	1	0
Flag	-	-	P3DIR5	P3DIR4	P3DIR3	-	-	-
At reset	-	-	0	0	0	-	-	-
Access	-	-	R/W	R/W	R/W	-	-	-

bp	Flag	Description
7-6	-	-
5-3	P3DIR5-3	I/O mode selection 0: Input mode 1: Output mode
2-0	-	-

■ Port 3 Pull-up/pull-down Resistor Control Register (P3PLUD: 0x03EA3)

MN101EFA8/MN101EFA3

bp	7	6	5	4	3	2	1	0
Flag	-	-	P3PLUD5	P3PLUD4	P3PLUD3	-	-	-
At reset	-	-	0	0	0	-	-	-
Access	-	-	R/W	R/W	R/W	-	-	-

bp	Flag	Description
7-6	-	-
5-3	P3PLUD5-3	Pull-up/pull-down resistor selection 0: Not added 1: Added
2-0	-	-

■ Port 3 Nch Open-drain Control Register (P3ODC: 0x03EF3)

MN101EFA8/MN101EFA3

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	P3ODC4	P3ODC3	-	-	-
At reset	-	-	-	0	0	-	-	-
Access	-	-	-	R/W	R/W	-	-	-

bp	Flag	Description
7-5	-	-
4	P3ODC4	P34 Nch open-drain output selection 0: Push-pull output 1: Nch open-drain output
3	P3ODC3	P33 Nch open-drain output selection 0: Push-pull output 1: Nch open-drain output
2-0	-	-

■ Port 3 Input Mode Register (P3IMD: 0x03EC3)

MN101EFA8/MN101EFA3

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	P3IMD3	-	-	-
At reset	-	-	-	-	0	-	-	-
Access	-	-	-	-	R/W	-	-	-

bp	Flag	Description
7-4	-	-
3	P3IMD3	I/O port or analog input selection 0: P33 1: AN15
2-0	-	-

■ Pull-up/pull-down Resistor Selection Register (SELUD: 0x03EAF)

MN101EFA8/MN101EFA3

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	SELUD3	-	-	-
At reset	-	-	-	-	0	-	-	-
Access	-	-	-	-	R/W	-	-	-

bp	Flag	Description
7-4	-	-
3	SELUD3	Pull-up/pull-down selection (Port 3) 0: Pull-up 1: Pull-down
2-0	-	-

5.5.3 Block Diagram

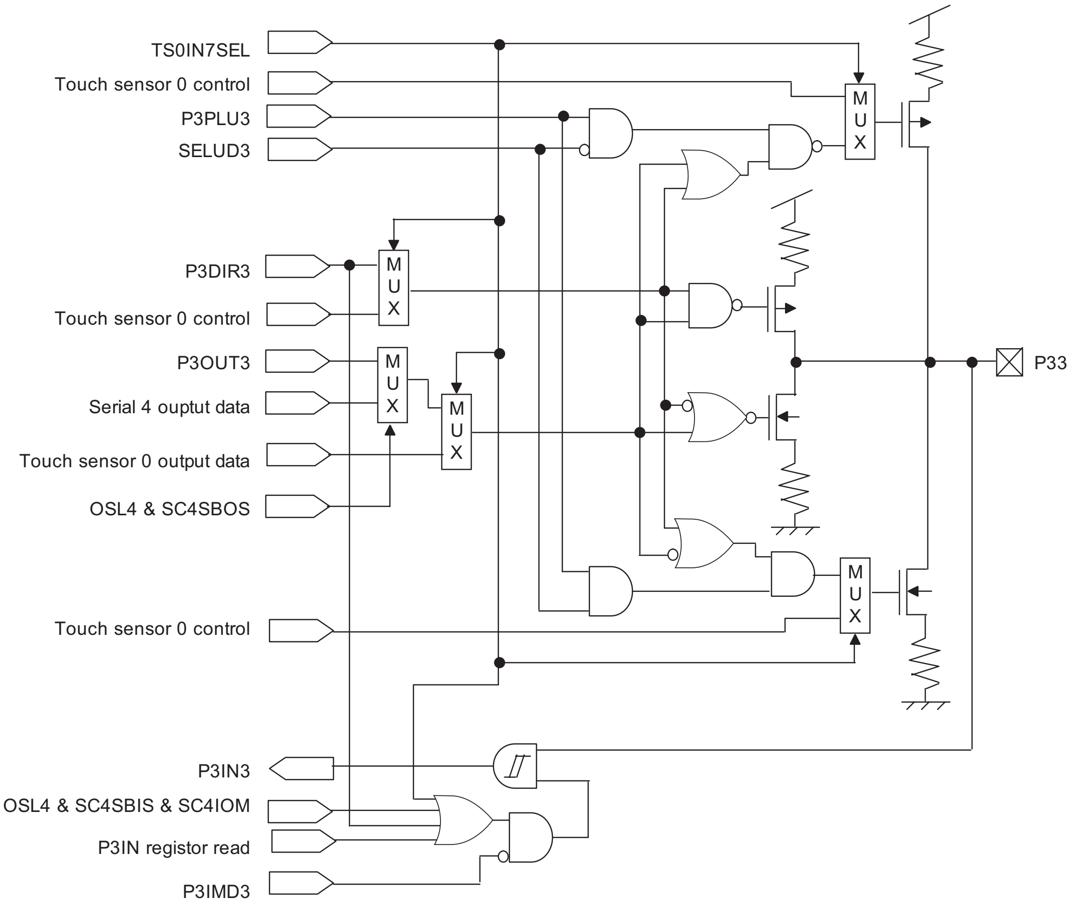


Figure:5.5.1 P33 Block Diagram (MN101EFA8)

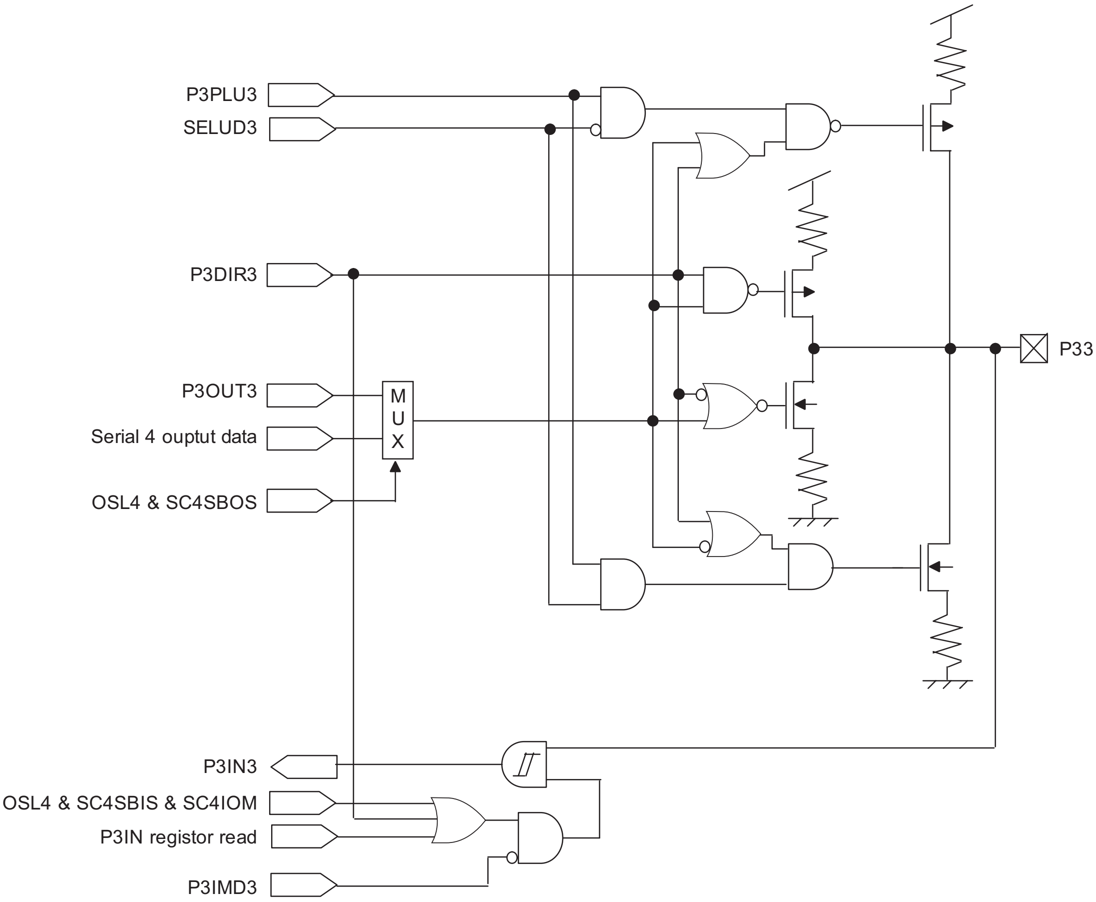


Figure:5.5.2 P33 Block Diagram (MN101EFA3)

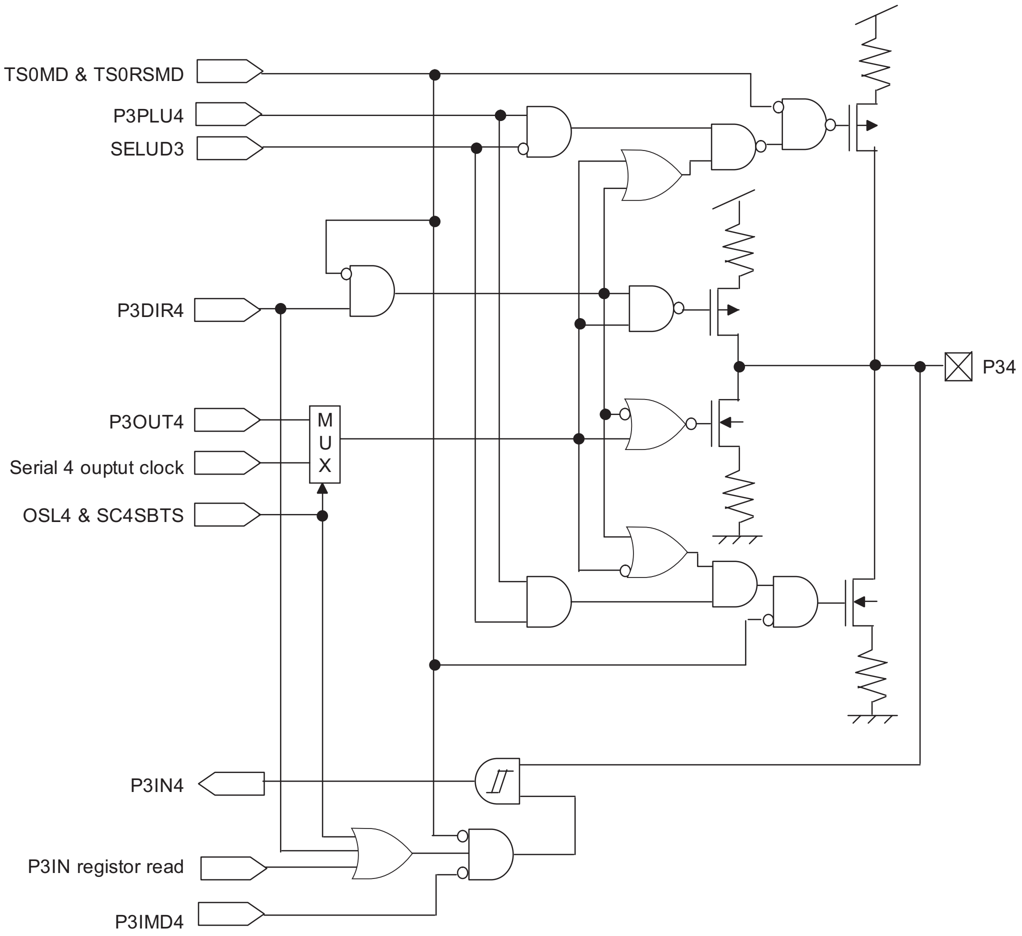


Figure:5.5.3 P34 Block Diagram (MN101EFA8)

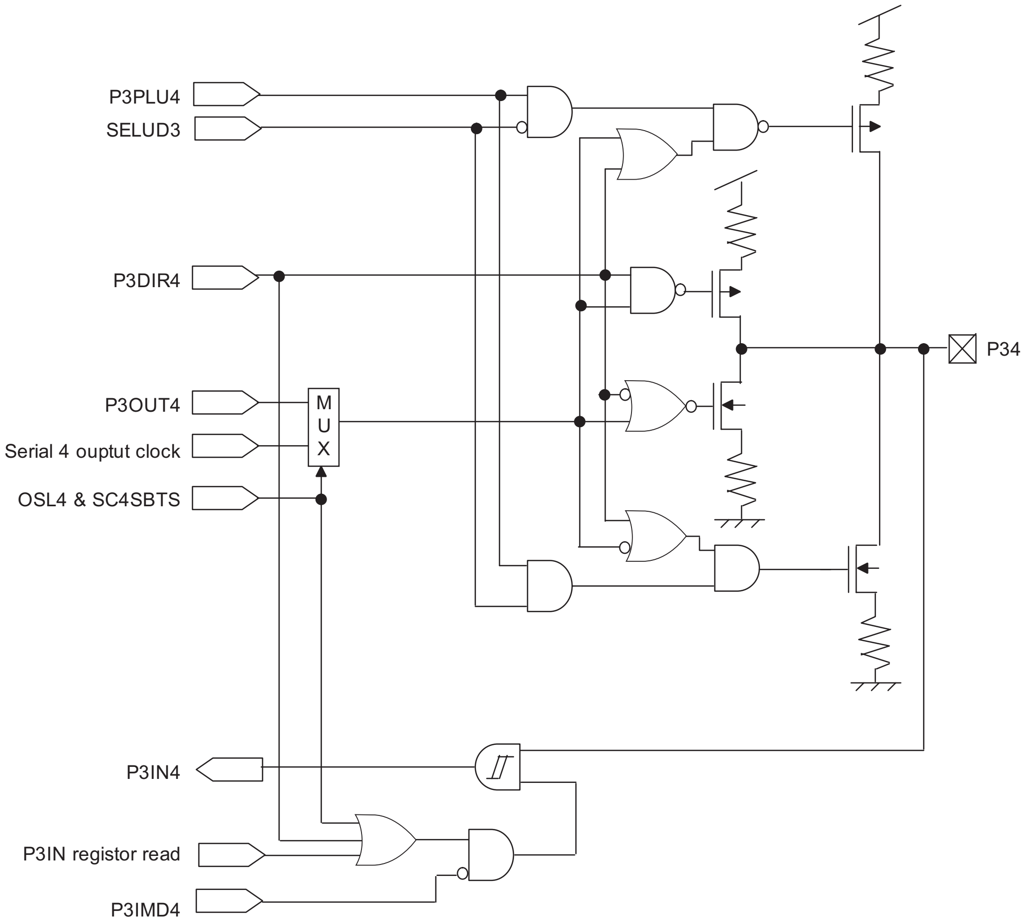


Figure:5.5.4 P34 Block Diagram (MN101EFA3)

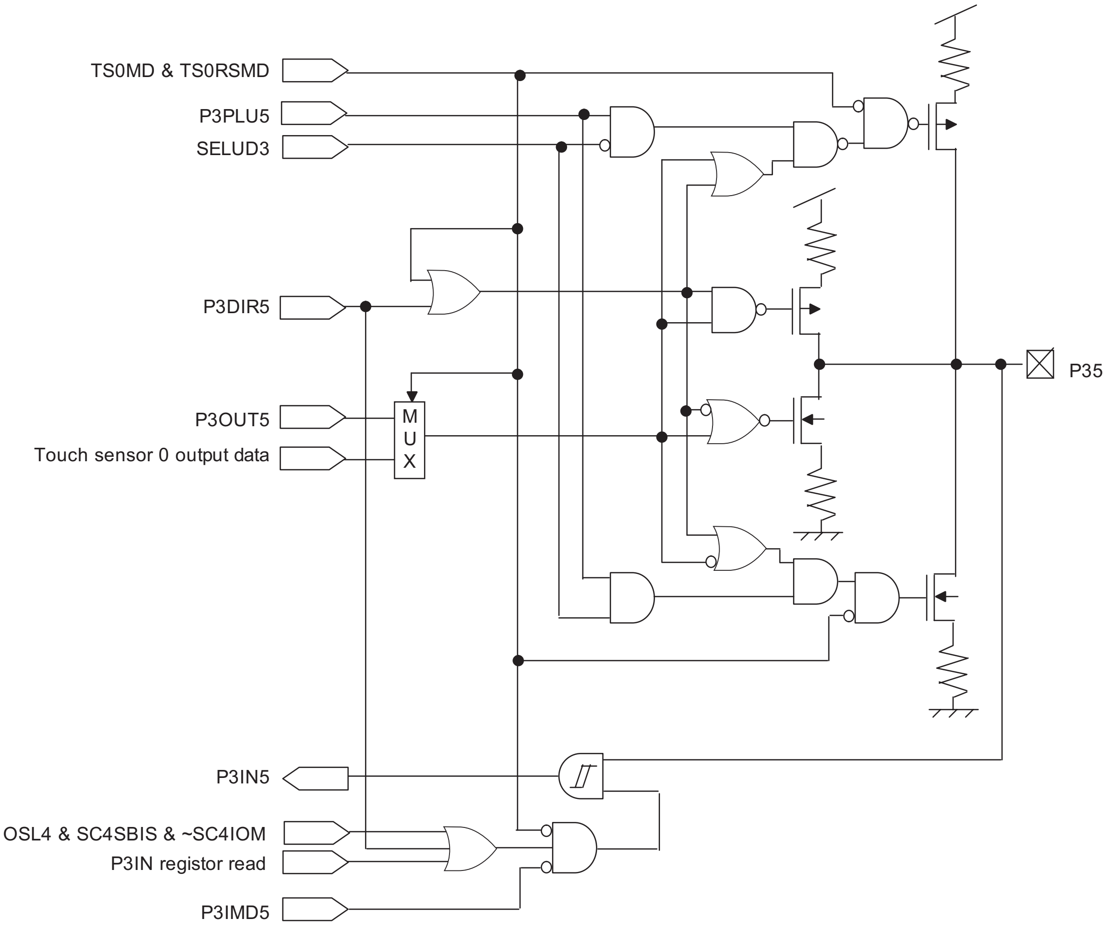


Figure:5.5.5 P35 Block Diagram (MN101EFA8)

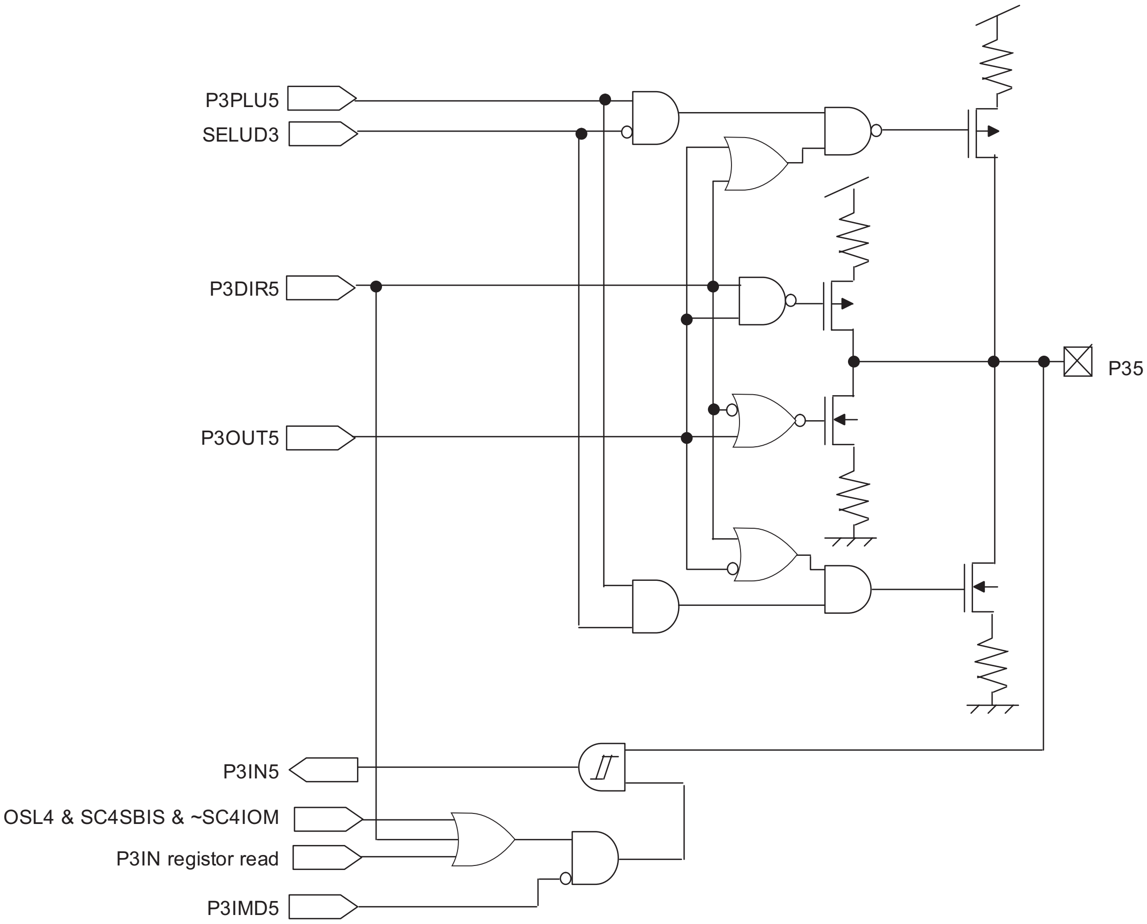


Figure:5.5.6 P35 Block Diagram (MN101EFA3)

5.6 Port 4

5.6.1 Description

MN101EFA7/A2 do not include the Port 4.

■ General Port Setup

To output data to pins, set the control flag of P4DIR register to "1" and write the data to P4OUT register.
To read input data of pins, set the control flag of P4DIR register to "0" and read the value of P4IN register.

Each bit can be set individually as either an input or output by P4DIR register. The control flag of P4DIR register is set to "1" for output mode, and "0" for input mode.

Each bit can be set individually if pull-up resistor is added or not by P4PLU register. Set the control flag of P4PLU register to "1" to add the pull-up resistor.

Each bit can be selected individually as Nch open-drain output by P4ODC register. P4ODC register is set to "1" for Nch open-drain output, "0" for push-pull output.

■ Special Function Pin Setup

P43 is also used as the I/O pin of serial 0 transmission/reception data and UART0 transmission data. When the SC0SBOS flag of SC0MD1 register is "1", P43 is the serial data I/O pin. Push-pull output or Nch open-drain output can be selected by setting P4ODC register.

P44 is also used as the input pin of serial 0 reception data and UART0 reception data. When the SC0SBIS flag of SC0MD1 register is "1", P44 is the serial data input pin.

P45 is also used as the I/O pin of serial 0 clock. When the SC0SBTS flag of the serial interface 0 mode register 1 (SC0MD1) is "1", P45 is the I/O pin of the serial clock. Push-pull output or Nch open-drain output can be selected by setting P4ODC register.

(This function is equipped in MN101EFA8.)
P45 to P47 are also used as input pins for touch sensor timer. Set "Used" to corresponding channel by TS1TCHSEL register. Refer to [Chapter XV Touch Sensor Timer].

Table:5.6.1 The port 4 Special Function Pins

Table remarks √: With function -: Without function

Pins	Special Functions	MN101EFA8	MN101EFA3	MN101EFA7	MN101EFA2
P43	SBO0B	√	√	-	-
	TXD0B	√	√	-	-
	TS1OP	√	-	-	-
P44	SBI0B	√	√	-	-
	RXD0B	√	√	-	-
	TS1RC	√	-	-	-
P45	SBT0B	√	√	-	-
	TS1IN0	√	-	-	-
P46	-	√	√	-	-
	TS1IN1	√	-	-	-
P47	-	√	√	-	-
	TS1IN2	√	-	-	-

5.6.2 Registers

■ Port 4 Output register (P4OUT: 0x03E74)

MN101EFA8/MN101EFA3

bp	7	6	5	4	3	2	1	0
Flag	P4OUT7	P4OUT6	P4OUT5	P4OUT4	P4OUT3	-	-	-
At reset	x	x	x	x	x	-	-	-
Access	R/W	R/W	R/W	R/W	R/W	-	-	-

bp	Flag	Description
7-3	P4OUT7-3	Output data 0: Output "Low" (V_{SS} level) 1: Output "High" (V_{DD5} level)
2-0	-	-

■ Port 4 Input Register (P4IN: 0x03E84)

MN101EFA8/MN101EFA3

bp	7	6	5	4	3	2	1	0
Flag	P4IN7	P4IN6	P4IN5	P4IN4	P4IN3	-	-	-
At reset	x	x	x	x	x	-	-	-
Access	R	R	R	R	R	-	-	-

bp	Flag	Description
7-3	P4IN7-3	Input data 0: Pin is "Low" (V_{SS} level) 1: Pin is "High" (V_{DD5} level)
2-0	-	-

■ Port 4 Direction Control Register (P4DIR: 0x03E94)

MN101EFA8/MN101EFA3

bp	7	6	5	4	3	2	1	0
Flag	P4DIR7	P4DIR6	P4DIR5	P4DIR4	P4DIR3	-	-	-
At reset	0	0	0	0	0	-	-	-
Access	R/W	R/W	R/W	R/W	R/W	-	-	-

bp	Flag	Description
7-3	P4DIR7-3	I/O mode selection 0: Input mode 1: Output mode
2-0	-	-

■ Port 4 Pull-up Resistor Control Register (P4PLU: 0x03EA4)

MN101EFA8/MN101EFA3

bp	7	6	5	4	3	2	1	0
Flag	P4PLU7	P4PLU6	P4PLU5	P4PLU4	P4PLU3	-	-	-
At reset	0	0	0	0	0	-	-	-
Access	R/W	R/W	R/W	R/W	R/W	-	-	-

bp	Flag	Description
7-3	P4PLU7-3	Pull-up/pull-down resistor selection 0: Not added 1: Added
2-0	-	-

■ Port 4 Nch open-drain Control Register (P4ODC: 0x03EF4)

MN101EFA8/A3

bp	7	6	5	4	3	2	1	0
Flag	-	-	P4ODC5	-	P4ODC3	-	-	-
At reset	-	-	0	-	0	-	-	-
Access	-	-	R/W	-	R/W	-	-	-

bp	Flag	Description
7-6	-	-
5	P4ODC5	P45 Nch open-drain output selection 0: Push-pull output 1: Nch open-drain output
-	-	-
3	P4ODC3	P43 Nch open-drain output selection 0: Push-pull output 1: Nch open-drain output
2-0	-	-

5.6.3 Block Diagram

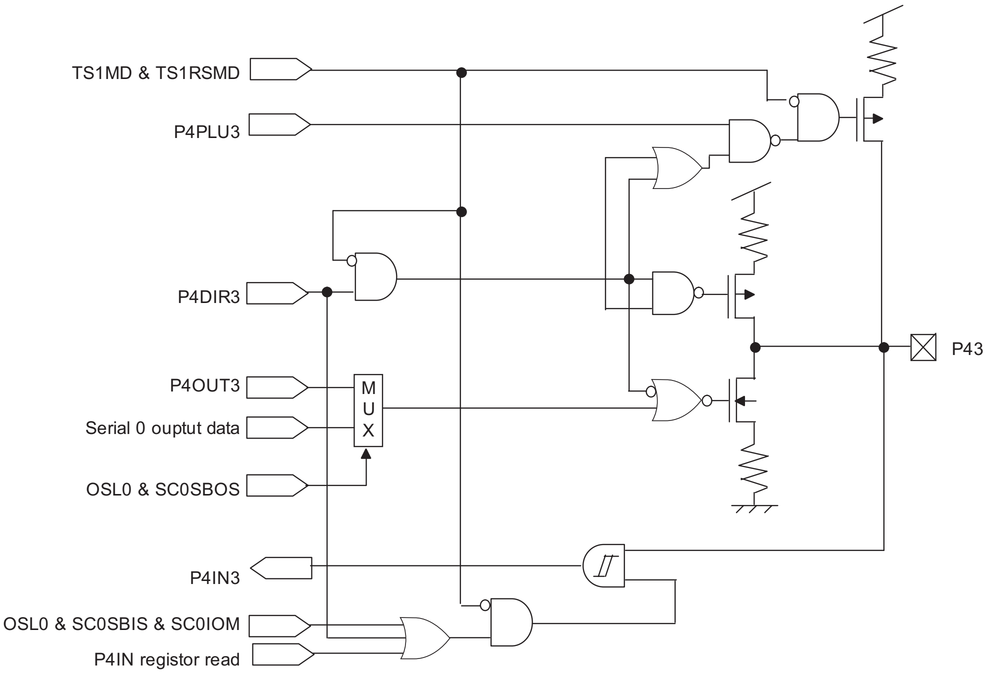


Figure:5.6.1 P43 Block Diagram (MN101EFA8)

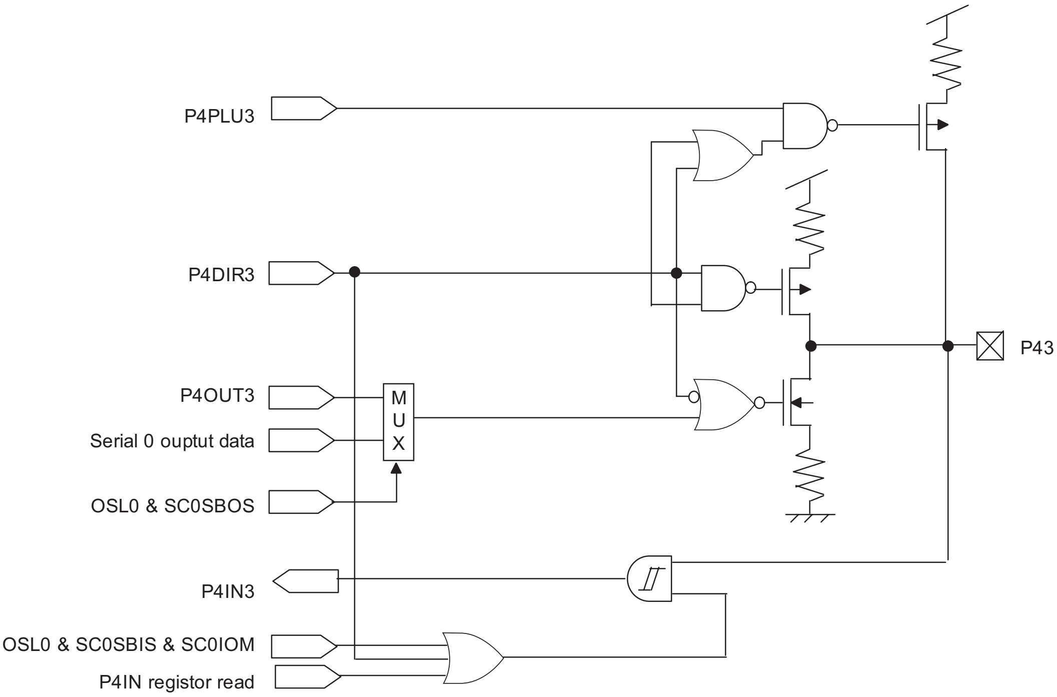


Figure:5.6.2 P43 Block Diagram (MN101EFA3)

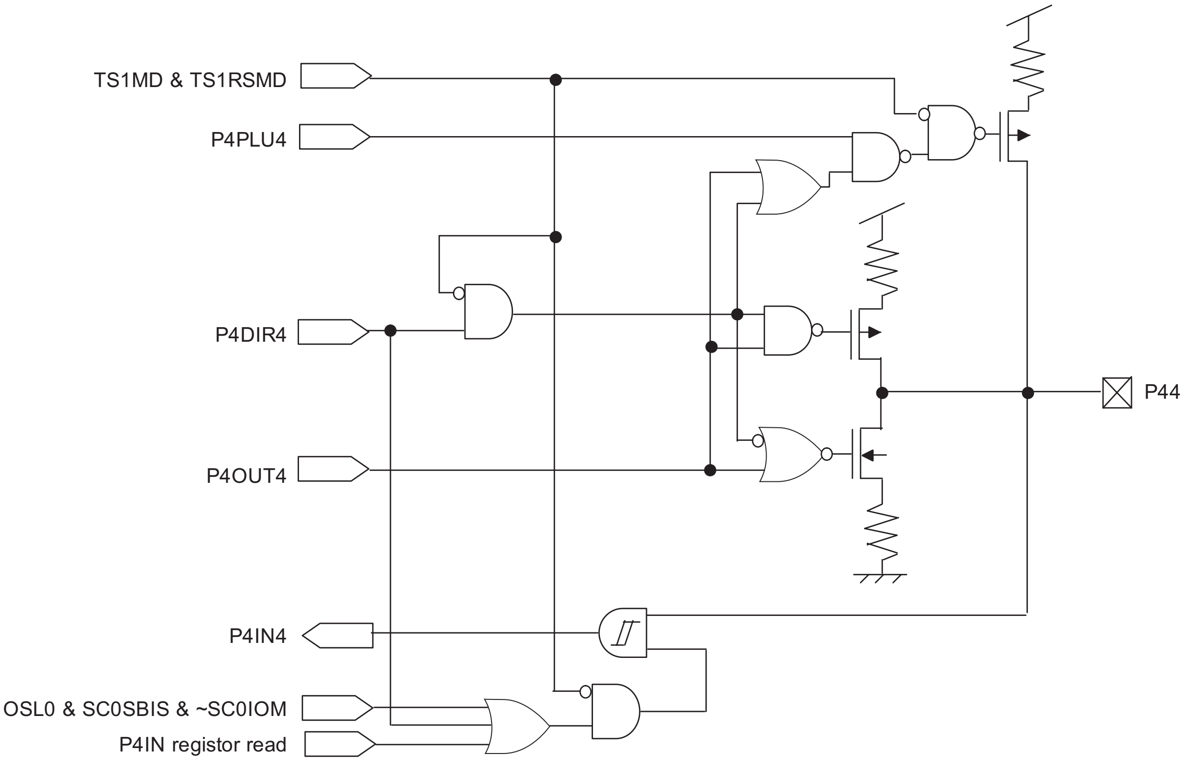


Figure:5.6.3 P44 Block Diagram (MN101EFA8)

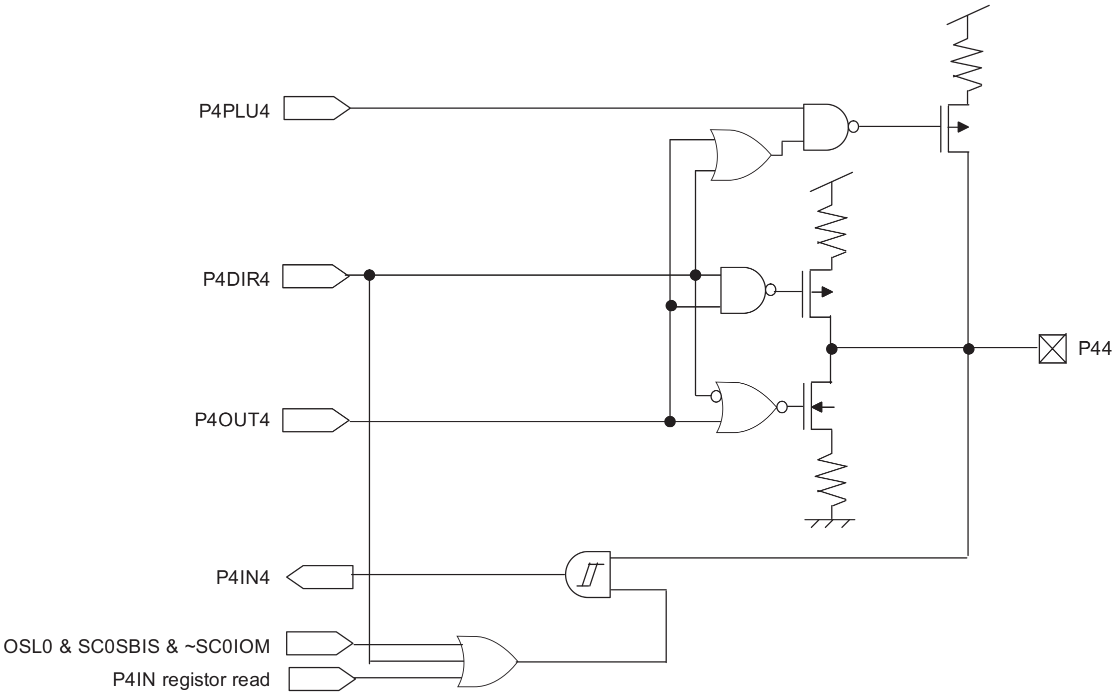


Figure:5.6.4 P44 Block Diagram (MN101EFA3)

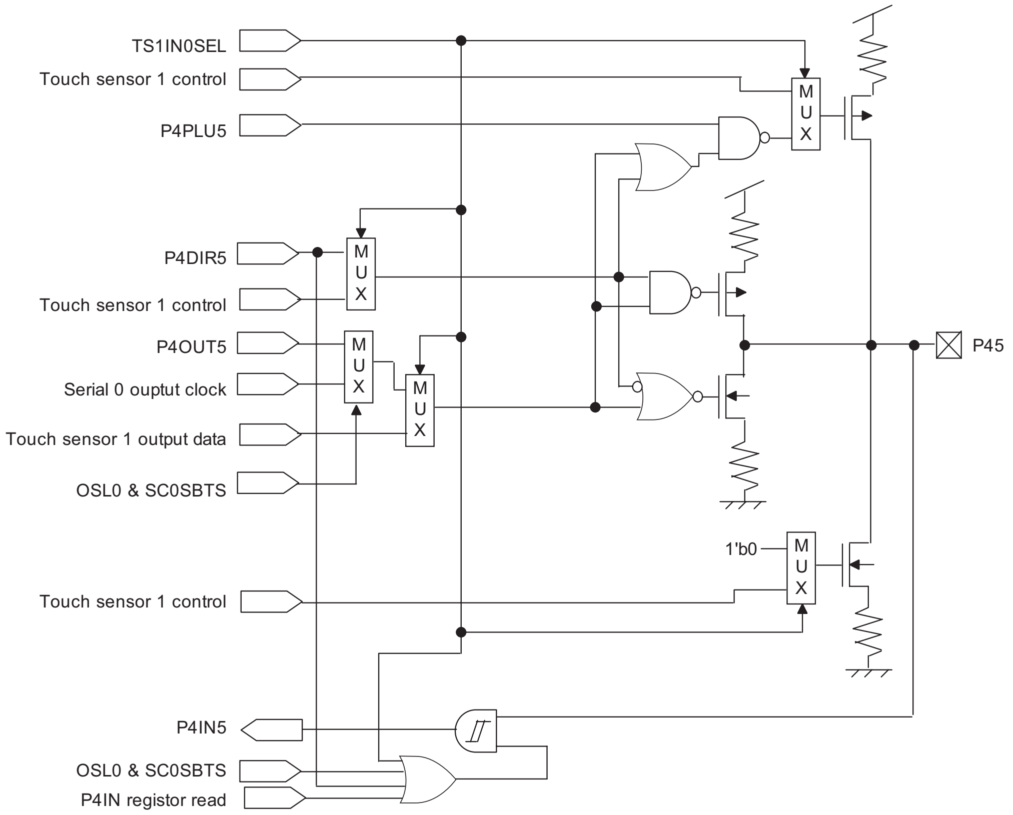


Figure:5.6.5 P45 Block Diagram (MN101EFA8)

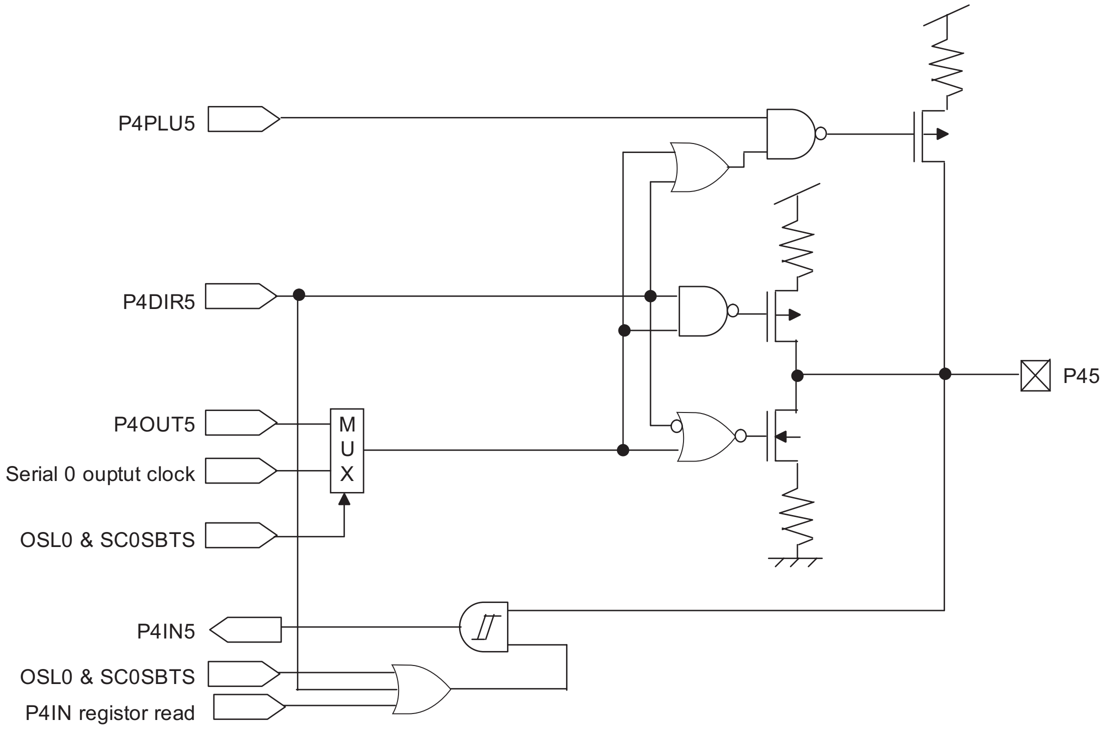


Figure:5.6.6 P45 Block Diagram (MN101EFA3)

5.7 Port 5

5.7.1 Description

■ General Pin Setup

To output data to pins, set the control flag of P5DIR register to "1" and write the data to P5OUT register.
To read input data of pin, set the control flag of P5DIR register to "0" and read the value of P5IN register.

Each bit can be set individually as either an input or output by P5DIR register. The control flag of P5DIR register is set to "1" for output mode, and "0" for input mode.

Each bit can be selected individually as Nch open-drain output by P5ODC register. P5ODC register is set to "1" for Nch open-drain output, "0" for push-pull output.

(This function is equipped in MN101EFA7/A2.)
Each bit can be selected individually as input mode by P5IMD register. P5IMD register is set to "1" to input the special function data and the value read from P5IN register is read to undefined, and "0" to use as the general port.

■ Special Function Pin Setup

P50 is also used as the I/O pin of serial 1 transmission/reception data and UART1 transmission data. When the SC1SBOS flag of SC1MD1 register is "1", P50 is the serial data I/O pin. Push-pull output or Nch open-drain output can be selected by setting P5ODC register.

P51 is also used as the input pin of serial 1 reception data and UART1 reception data. When the SC1SBIS flag of SC1MD1 register is "1", P51 is the serial data input pin.

P52 is also used as the I/O pin of serial 1 clock. When the SC1SBTS flag of the serial interface 1 mode register 1 (SC1MD1) is "1", P52 is the I/O pin of the serial clock. Push-pull output or Nch open-drain output can be selected by setting P5ODC register.

P56 is also used as the I/O pin of reverse buzzer.
P57 is also used as the I/O pin of buzzer.

Each bit for the I/O mode can be set individually by P5OMD register.
These registers are set to "1" to output the special function data, and "0" to be used as the general port.

(This function is equipped in MN101EFA8.)
P57 is also used as input pins for touch sensor timer.
Set "Used" to corresponding channel by TS1TCHSEL register. Refer to [Chapter XV Touch Sensor Timer].

(This function is equipped in MN101EFA7.)
P55 to P57 are also used as analog input pin. Input mode for each bit can be selected by P5IMD register.
When these pins are used as analog input pin, the value read from the port 5 input register is read to be "0".
P51 to P57 are also used as input pins for touch sensor timer.
Set "Used" to corresponding channel by TS0TCHSEL register. Refer to [Chapter XV Touch Sensor Timer].

(This function is equipped in MN101EFA2.)
P55 to P57 are also used as analog input pin. Input mode for each bit can be selected by P5IMD register.
When these pins are used as analog input pin, the value read from the port 5 input register is read to be "0".

Table:5.7.1 The port 5 Special Function Pins

Table remarks √: With function -: Without function

Pins	Special Functions	MN101EFA8	MN101EFA3	MN101EFA7	MN101EFA2
P50	SBO1A	√	√	√	√
	TXD1A	√	√	√	√
	TS0RC	-	-	√	-
P51	SBI1A	√	√	√	√
	RXD1A	√	√	√	√
	TS0IN7	-	-	√	-
P52	SBT1A	√	√	√	√
	TS0IN6	-	-	√	-
P53	-	√	√	√	√
	TS0IN5	-	-	√	-
P54	-	√	√	√	√
	AN11	-	-	√	√
	TS0IN4	-	-	√	-
P55	-	√	√	√	√
	AN10	-	-	√	√
	TS0IN3	-	-	√	-
P56	NBUZZERA	√	√	√	√
	AN9	-	-	√	√
	TS0IN2	-	-	√	-
P57	BUZZERA	√	√	√	√
	TS1IN3	√	-	-	-
	TS0IN1	-	-	√	-

5.7.2 Registers

- Port 5 Output Register (P5OUT: 0x03E75)

bp	7	6	5	4	3	2	1	0
Flag	P5OUT7	P5OUT6	P5OUT5	P5OUT4	P5OUT3	P5OUT2	P5OUT1	P5OUT0
At reset	X	X	X	X	X	X	X	X
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-0	P5OUT7-0	Output data 0: Output "Low" (V_{SS} level) 1: Output "High" (V_{DD5} level)

- Port 5 Input Register (P5IN: 0x03E85)

bp	7	6	5	4	3	2	1	0
Flag	P5IN7	P5IN6	P5IN5	P5IN4	P5IN3	P5IN2	P5IN1	P5IN0
At reset	X	X	X	X	X	X	X	X
Access	R	R	R	R	R	R	R	R

bp	Flag	Description
7-0	P5IN7-0	Input data 0: Pin is "Low" (V_{SS} level) 1: Pin is "High" (V_{DD5} level)

- Port 5 Direction Control Register (P5DIR: 0x03E95)

bp	7	6	5	4	3	2	1	0
Flag	P5DIR7	P5DIR6	P5DIR5	P5DIR4	P5DIR3	P5DIR2	P5DIR1	P5DIR0
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-0	P5DIR7-0	I/O mode selection 0: Input mode 1: Output mode

■ Port 5 Pull-up Resistor Control Register (P5PLU: 0x03EA5)

MN101EFA8/MN101EFA3

bp	7	6	5	4	3	2	1	0
Flag	P5PLU7	P5PLU6	P5PLU5	P5PLU4	P5PLU3	P5PLU2	P5PLU1	P5PLU0
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-0	P5PLU7-0	Pull-up resistor selection 0: Not added 1: Added

■ Port 5 Pull-up/pull-down Resistor Control Register (P5PLUD: 0x03EA5)

MN101EFA7/MN101EFA2

bp	7	6	5	4	3	2	1	0
Flag	P5PLUD7	P5PLUD6	P5PLUD5	P5PLUD4	P5PLUD3	P5PLUD2	P5PLUD1	P5PLUD0
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-0	P5PLUD7-0	Pull-up/pull-down resistor selection 0: Not added 1: Added

■ Port 5 Output Mode Register (P5OMD: 0x03EB5)

bp	7	6	5	4	3	2	1	0
Flag	P5OMD7	P5OMD6	-	-	-	-	-	-
At reset	0	0	-	-	-	-	-	-
Access	R/W	R/W	-	-	-	-	-	-

bp	Flag	Description
7	P5OMD7	I/O port or special function selection 0: P57 1: BUZZERA
6	P5OMD6	I/O port or special function selection 0: P56 1: NBUZZERA
5-0	-	-

■ Port 5 Input Mode Register (P5IMD: 0x03EC5)

MN101EFA7/MN101EFA2

bp	7	6	5	4	3	2	1	0
Flag	P5IMD7	P5IMD6	P5IMD5	-	-	-	-	-
At reset	0	0	0	-	-	-	-	-
Access	R/W	R/W	R/W	-	-	-	-	-

bp	Flag	Description
7	P5IMD7	I/O port or analog input selection 0: P57 1: AN9
6	P5IMD6	I/O port or analog input selection 0: P56 1: AN10
5	P5IMD5	I/O port or analog input selection 0: P55 1: AN11
4-0	-	-

■ Port 5 Nch Open-drain Control Register (P5ODC: 0x03EF5)

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	-	P5ODC2	-	P5ODC0
At reset	-	-	-	-	-	0	-	0
Access	-	-	-	-	-	R/W	-	R/W

bp	Flag	Description
7-3	-	-
2	P5ODC2	P52 Nch open-drain output selection 0: Push-pull output 1: Nch open-drain output
1	-	-
0	P5ODC0	P50 Nch open-drain output selection 0: Push-pull output 1: Nch open-drain output

■ Port 5 Pull-up/pull-down Resistor Selection Register (SELUD: 0x03EAF)

MN101EFA7/MN101EFA2

bp	7	6	5	4	3	2	1	0
Flag	-	-	SELUD5	-	-	-	-	-
At reset	-	-	0	-	-	-	-	-
Access	-	-	R/W	-	-	-	-	-

bp	Flag	Description
7-6	-	-
5	SELUD5	Pull-up/pull-down selection (Port 5) 0: pull-up 1: pull-down
4-0	-	-

5.7.3 Block Diagram

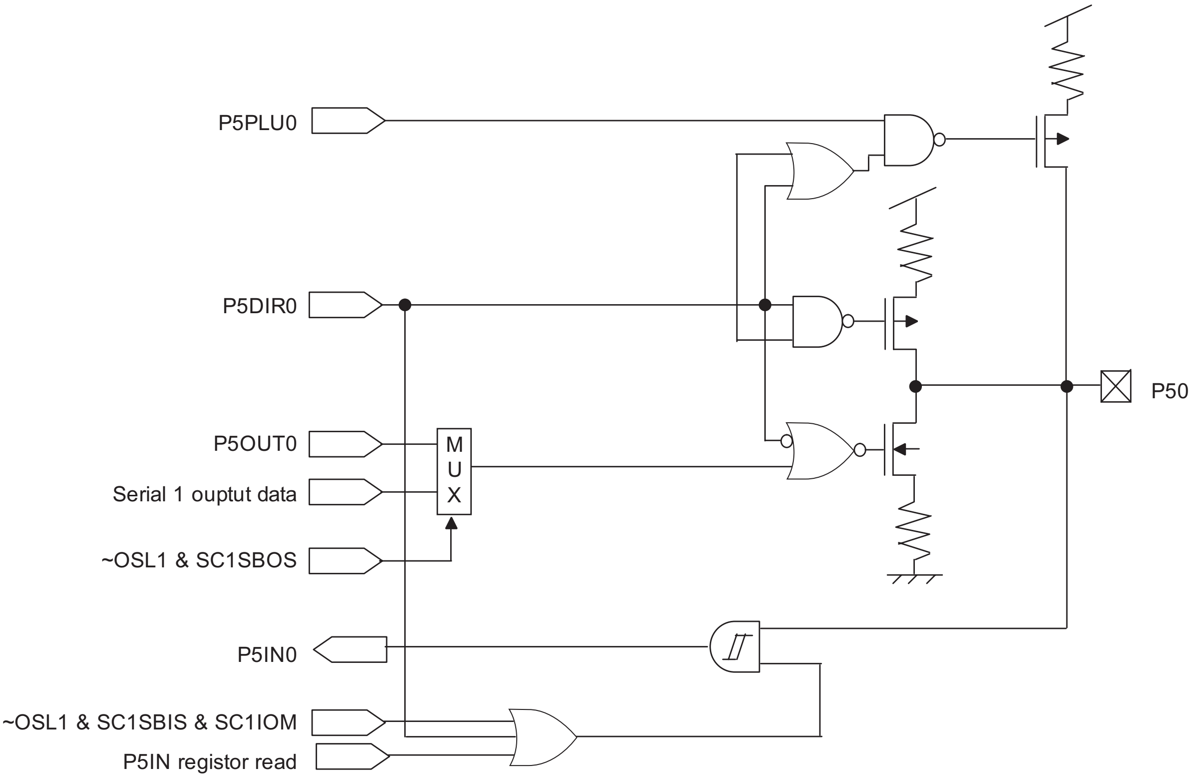


Figure:5.7.1 P50 Block Diagram (MN101EFA8/A3)

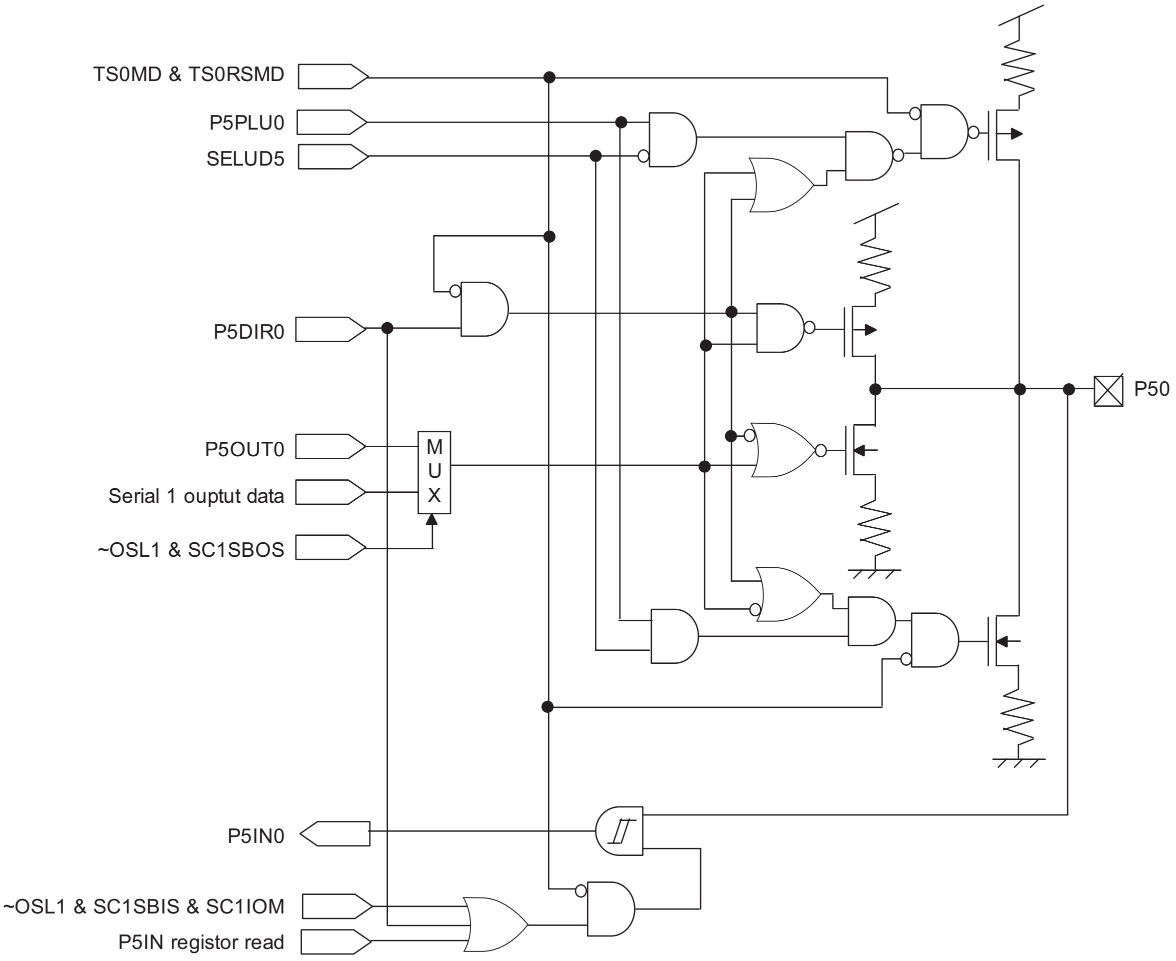


Figure:5.7.2 P50 Block Diagram (MN101EFA7)

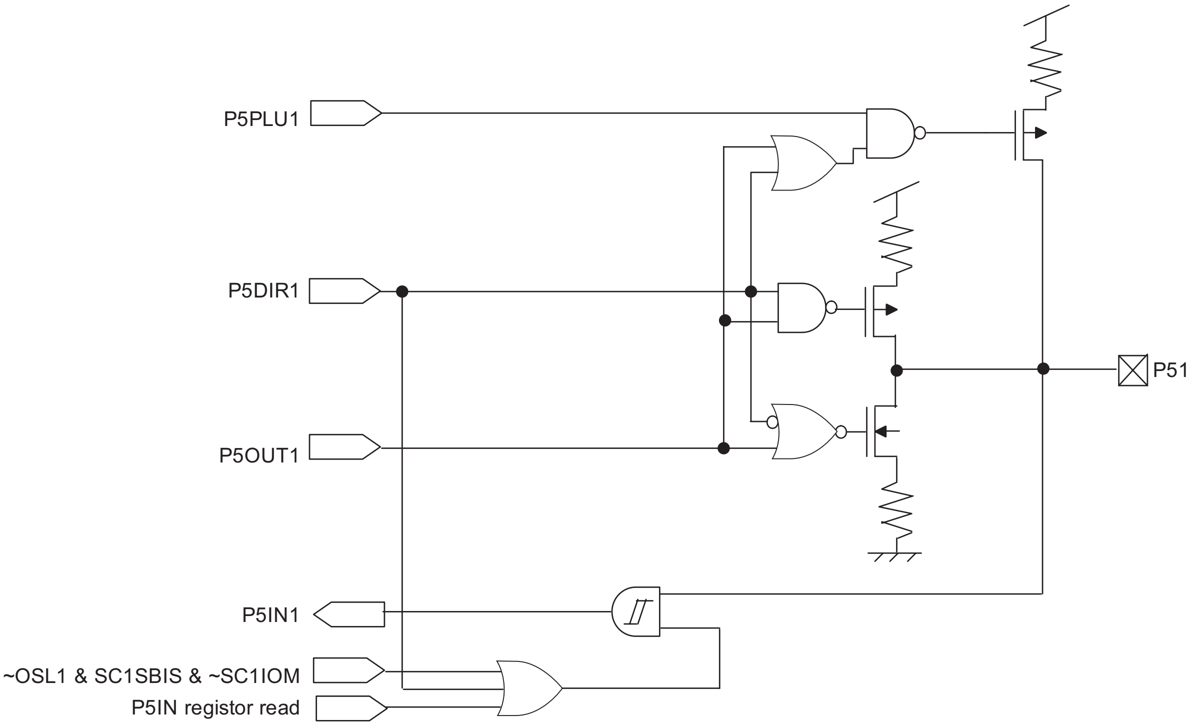


Figure:5.7.3 P51 Block Diagram (MN101EFA8/A3)

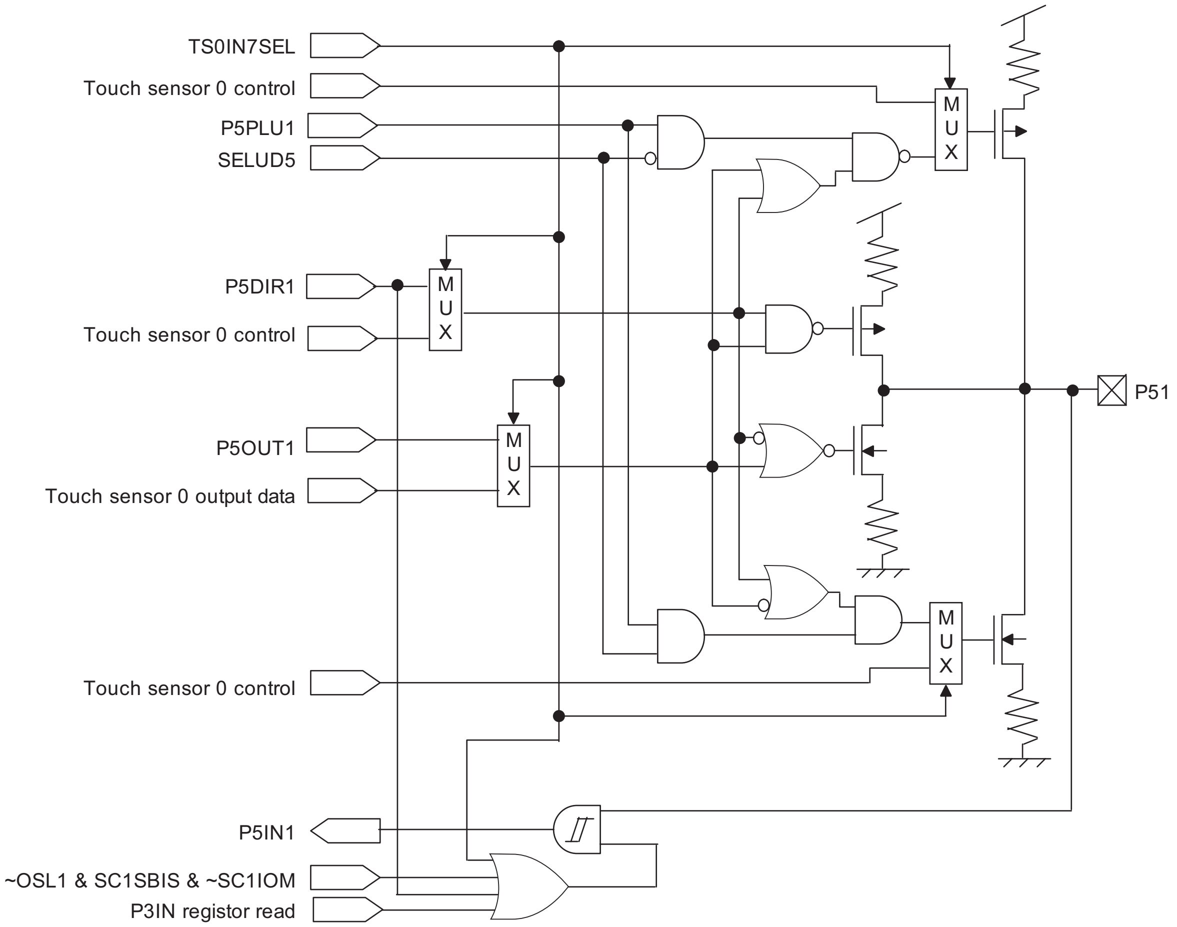


Figure:5.7.4 P51 Block Diagram (MN101EFA7)

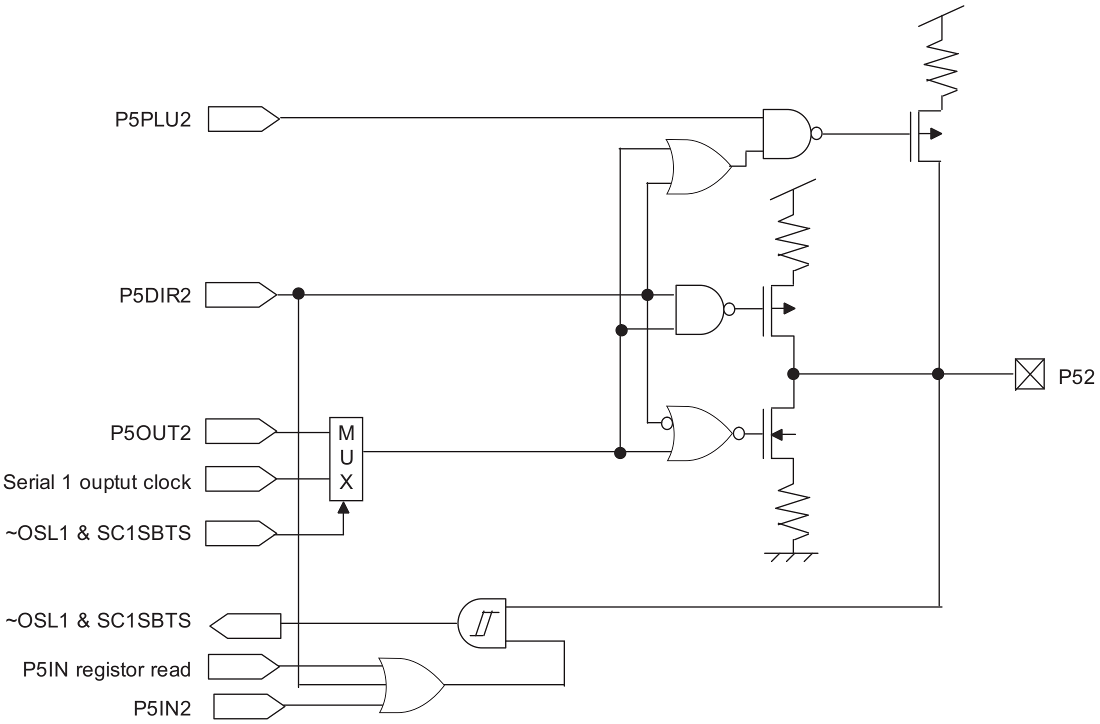


Figure:5.7.5 P52 Block Diagram (MN101EFA8/A3)

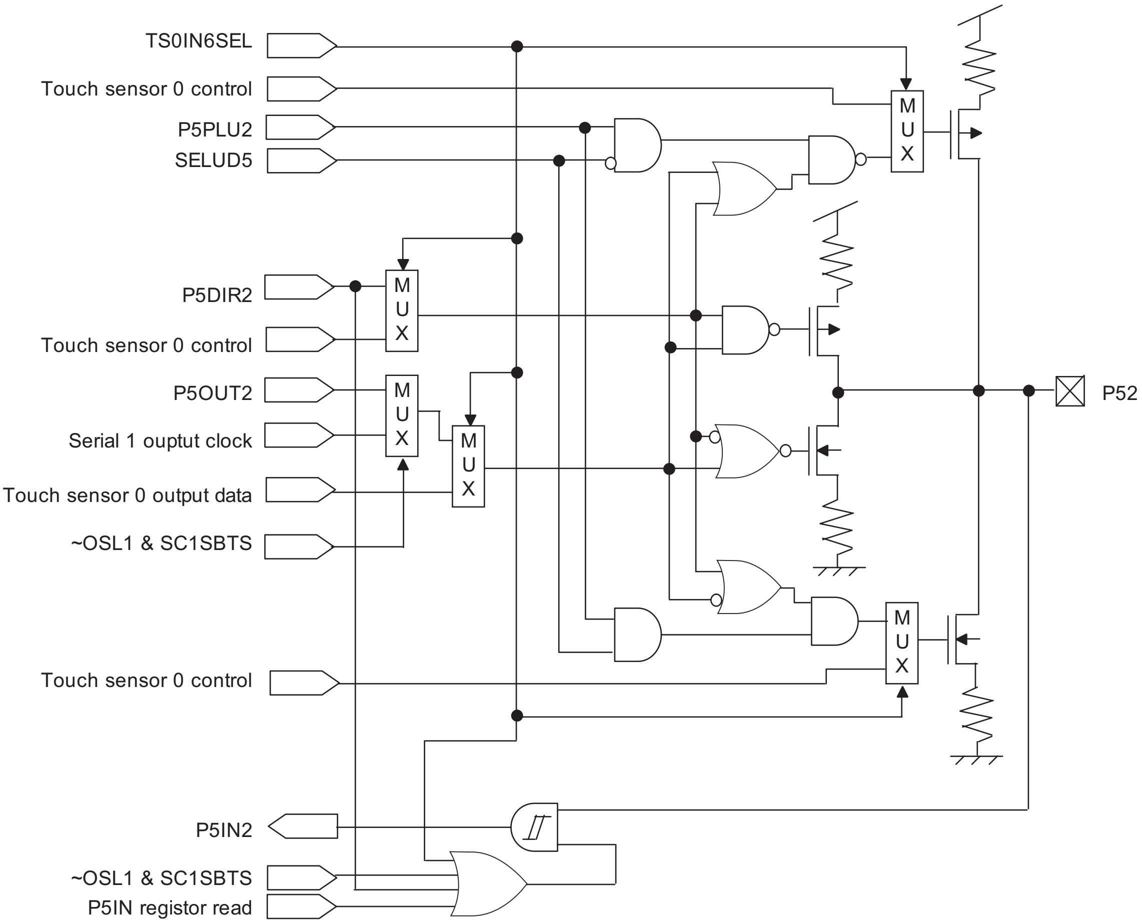


Figure:5.7.6 P52 Block Diagram (MN101EFA7)

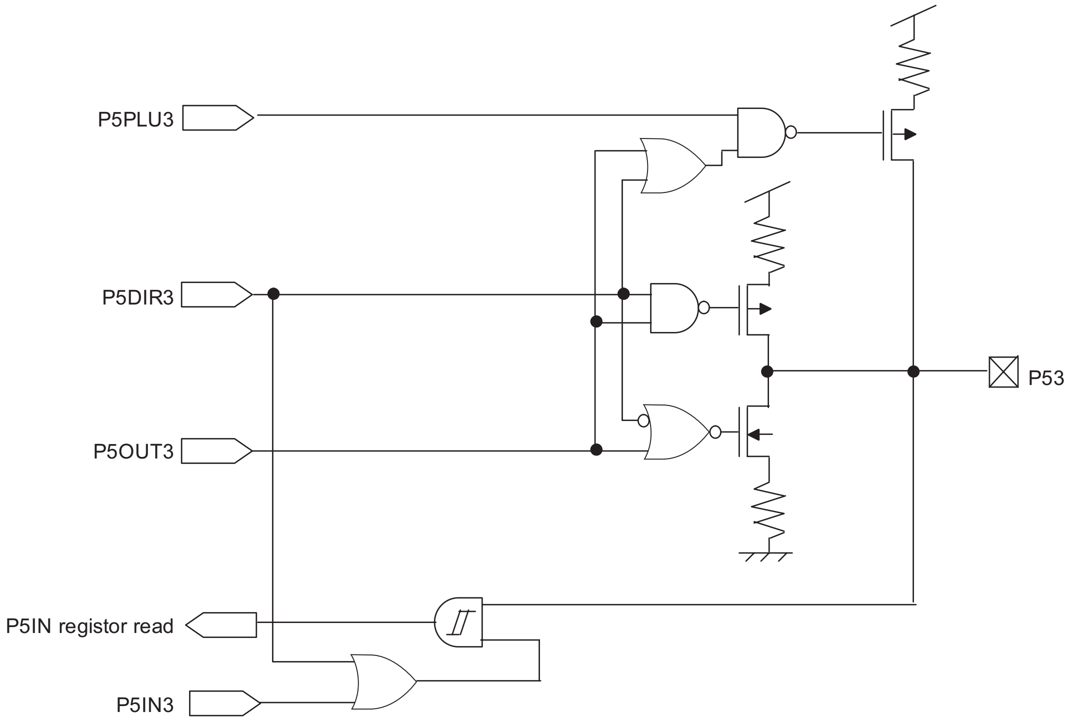


Figure:5.7.7 P53 Block Diagram (MN101EFA8/A3)

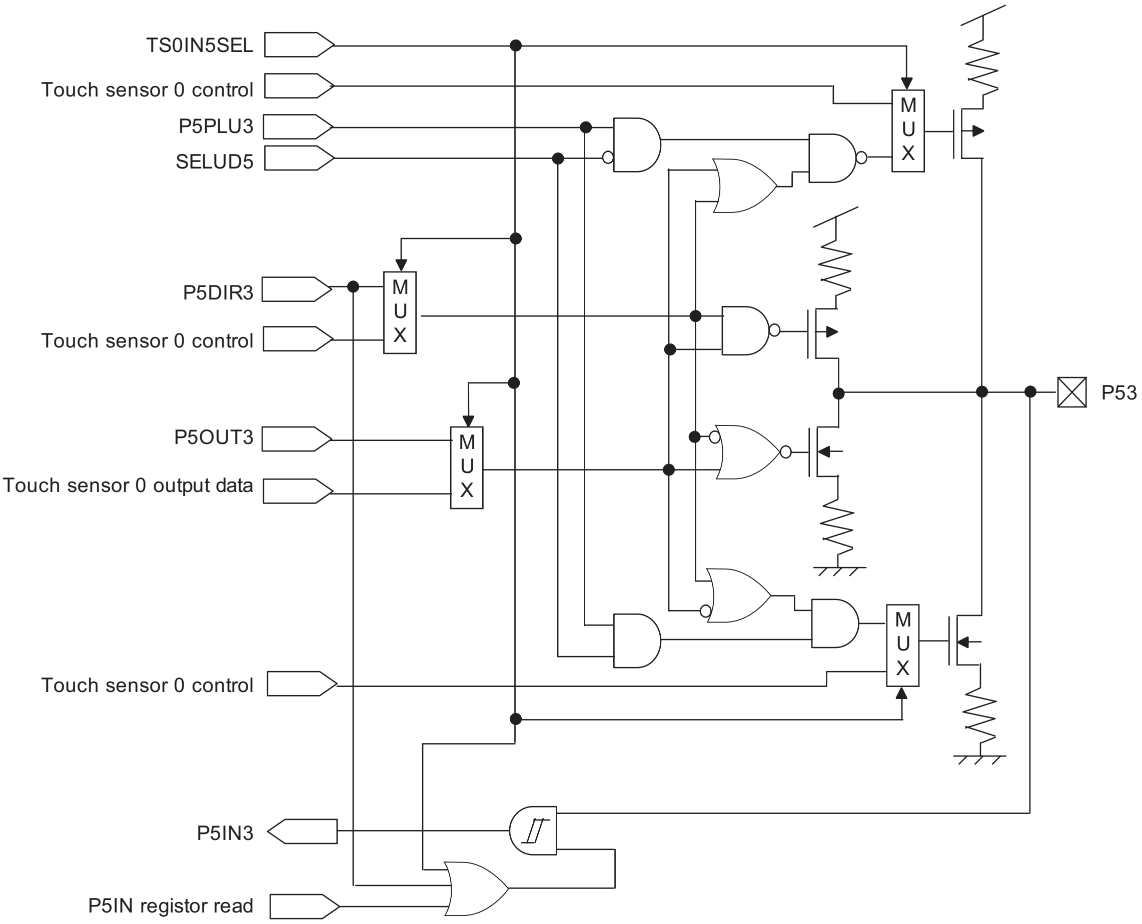


Figure:5.7.8 P53 Block Diagram (MN101EFA7)

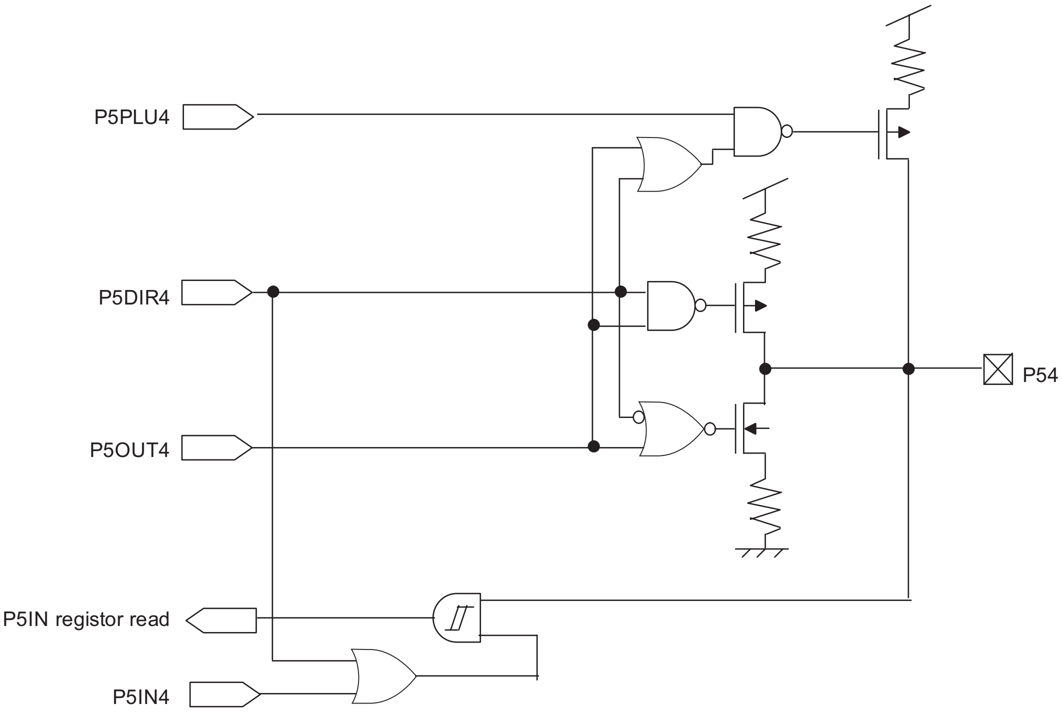


Figure:5.7.9 P54 Block Diagram (MN101EFA8/A3)

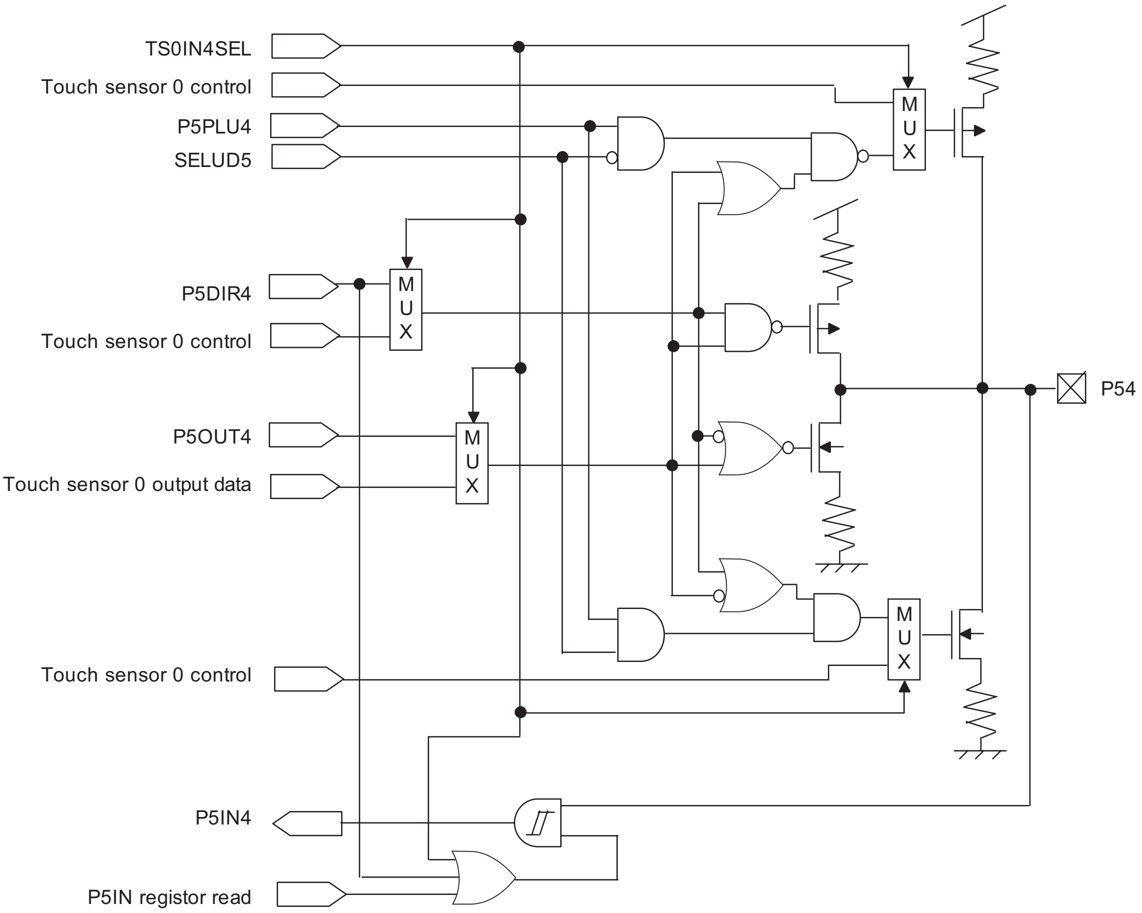


Figure:5.7.10 P54 Block Diagram (MN101EFA7)

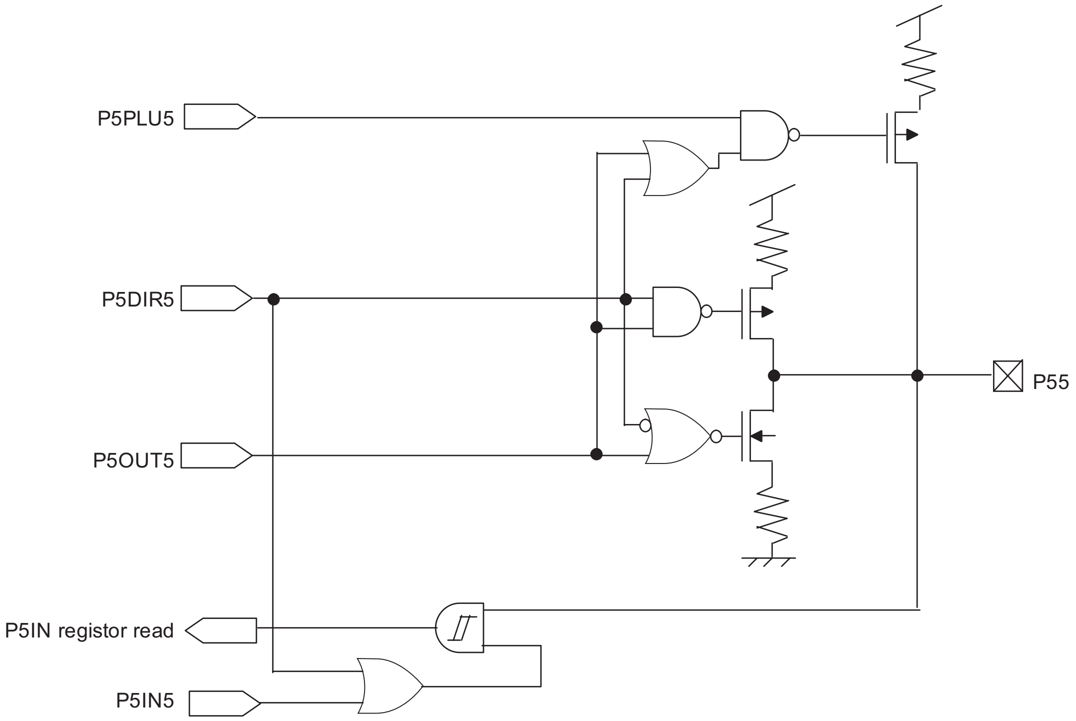


Figure:5.7.11 P55 Block Diagram (MN101EFA8/A3)

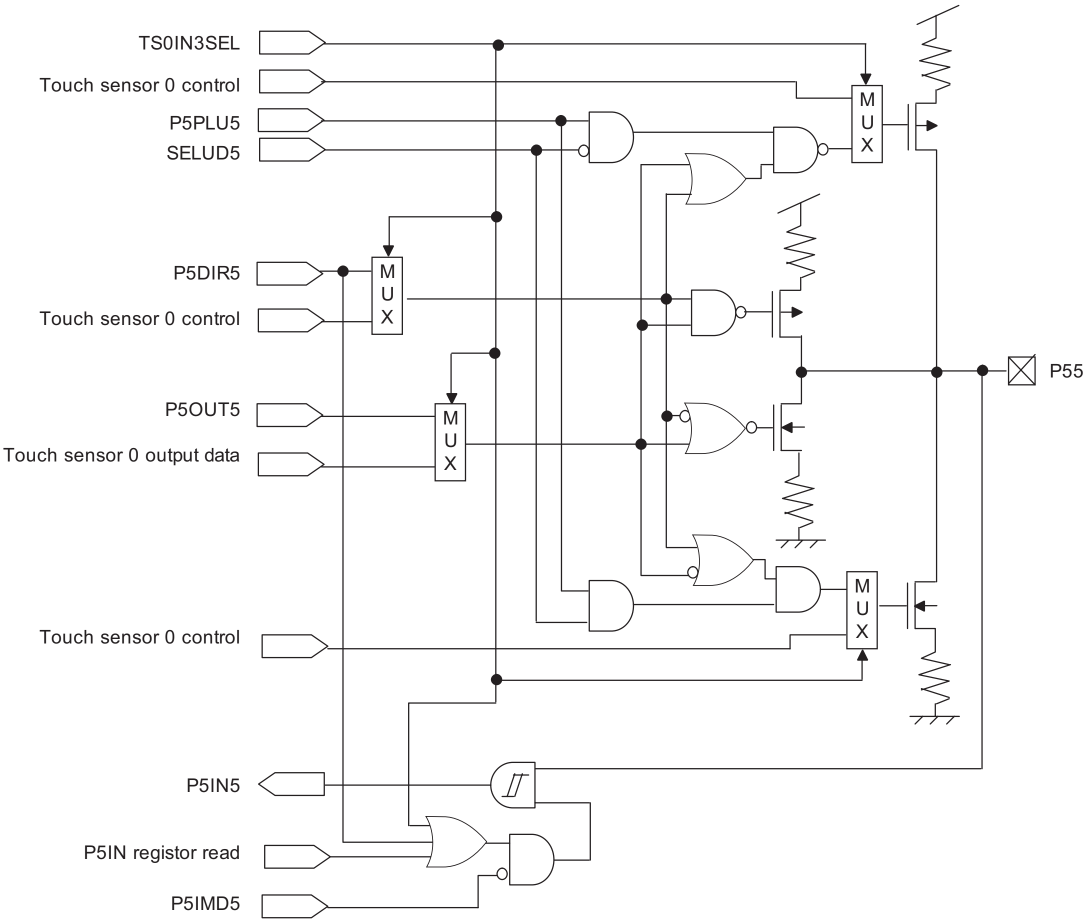


Figure:5.7.12 P55 Block Diagram (MN101EFA7)

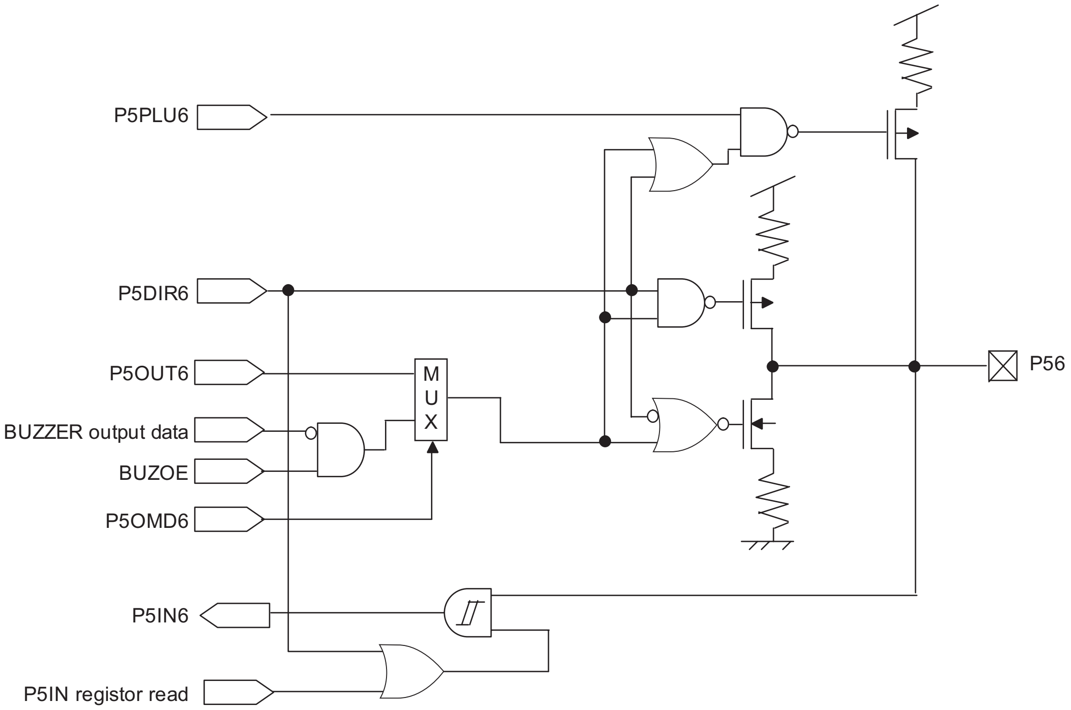


Figure:5.7.13 P56 Block Diagram (MN101EFA8/A3)

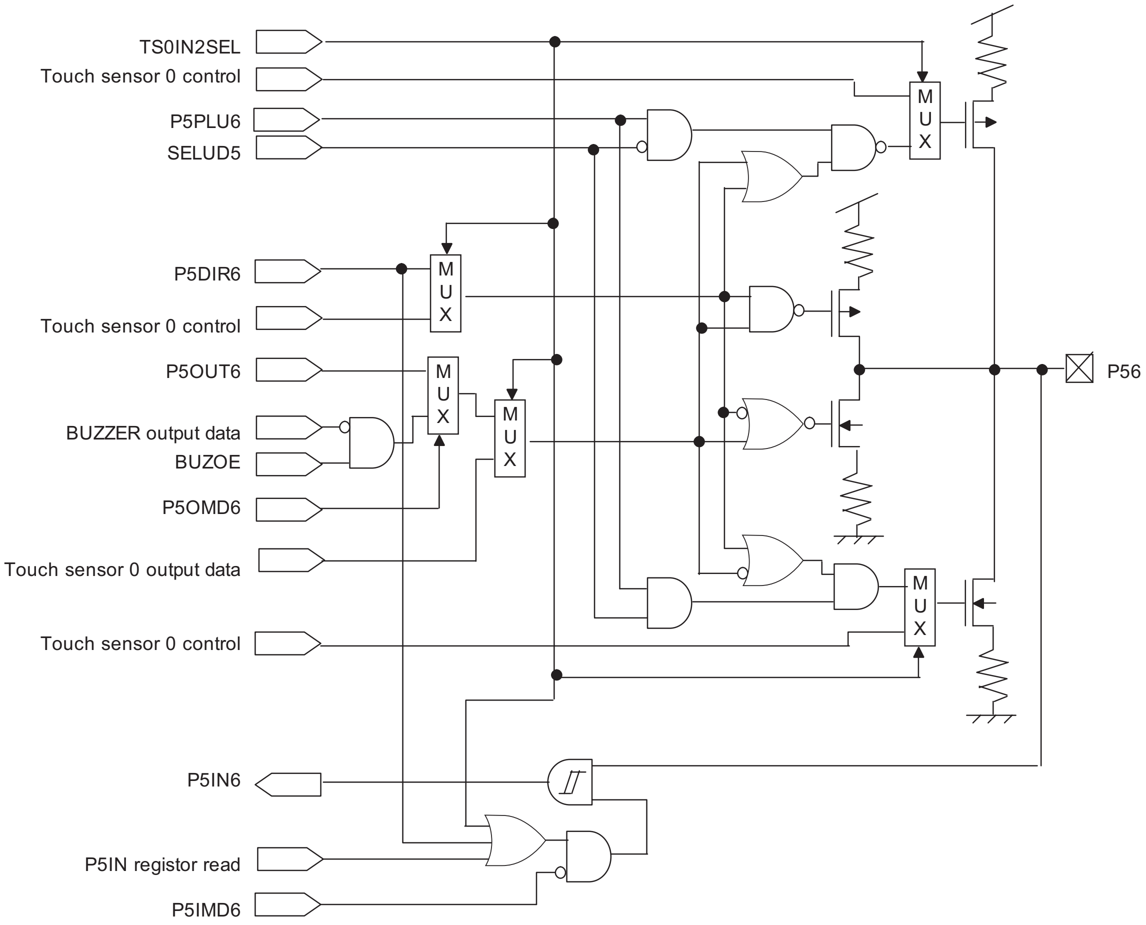


Figure:5.7.14 P56 Block Diagram (MN101EFA7)

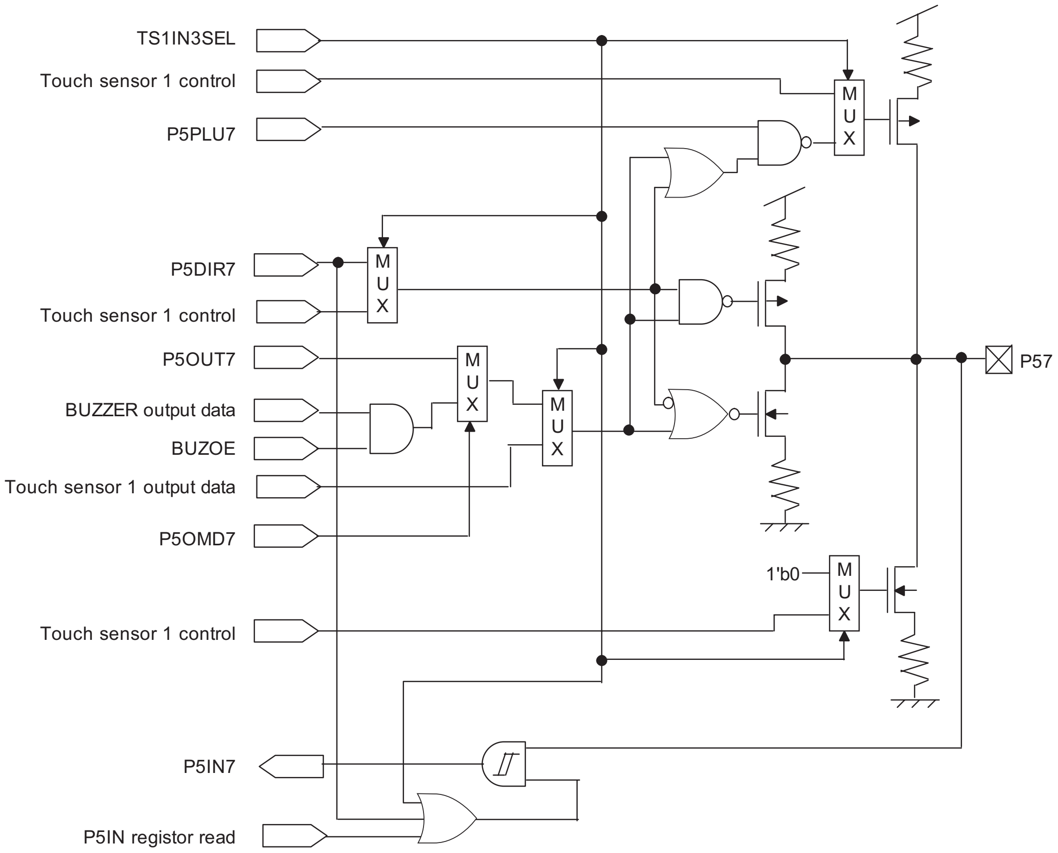


Figure:5.7.15 P57 Block Diagram (MN101EFA8)

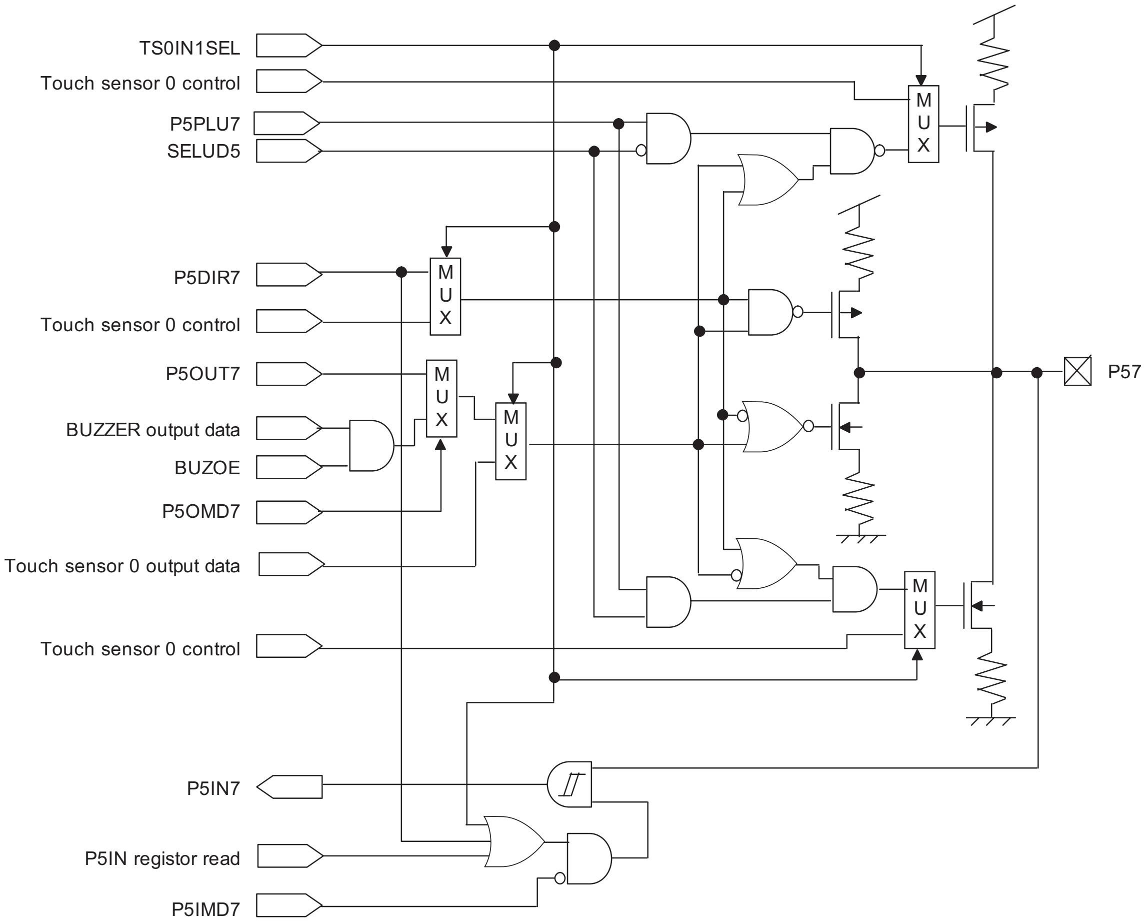


Figure:5.7.16 P57 Block Diagram (MN101EFA7)

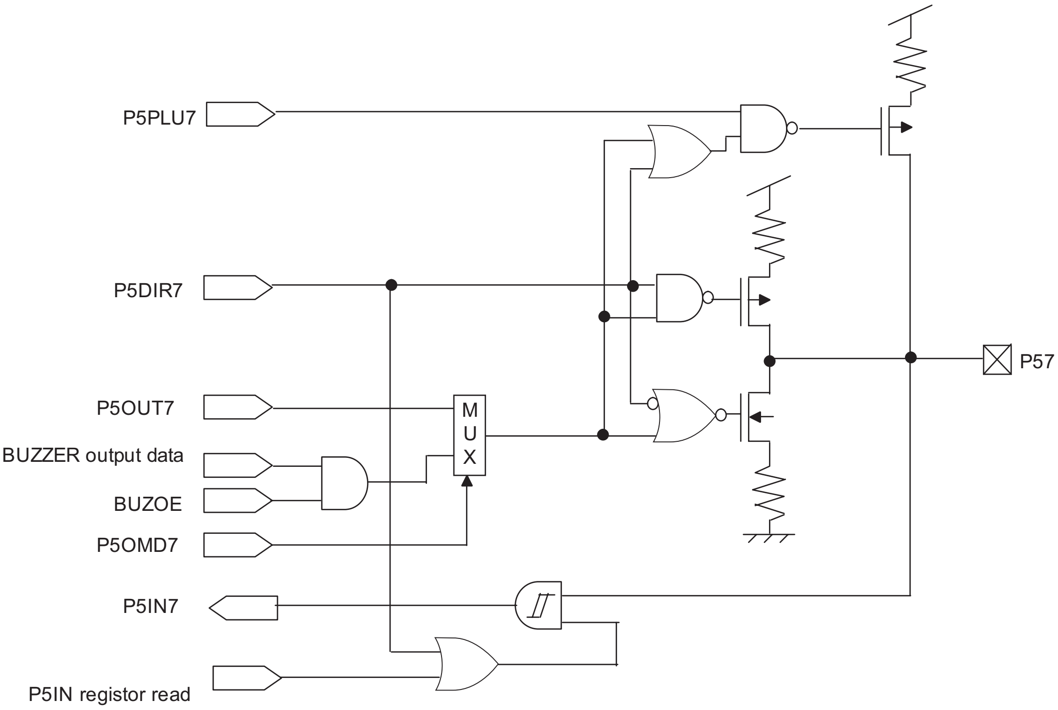


Figure:5.7.17 P57 Block Diagram (MN101EFA3)

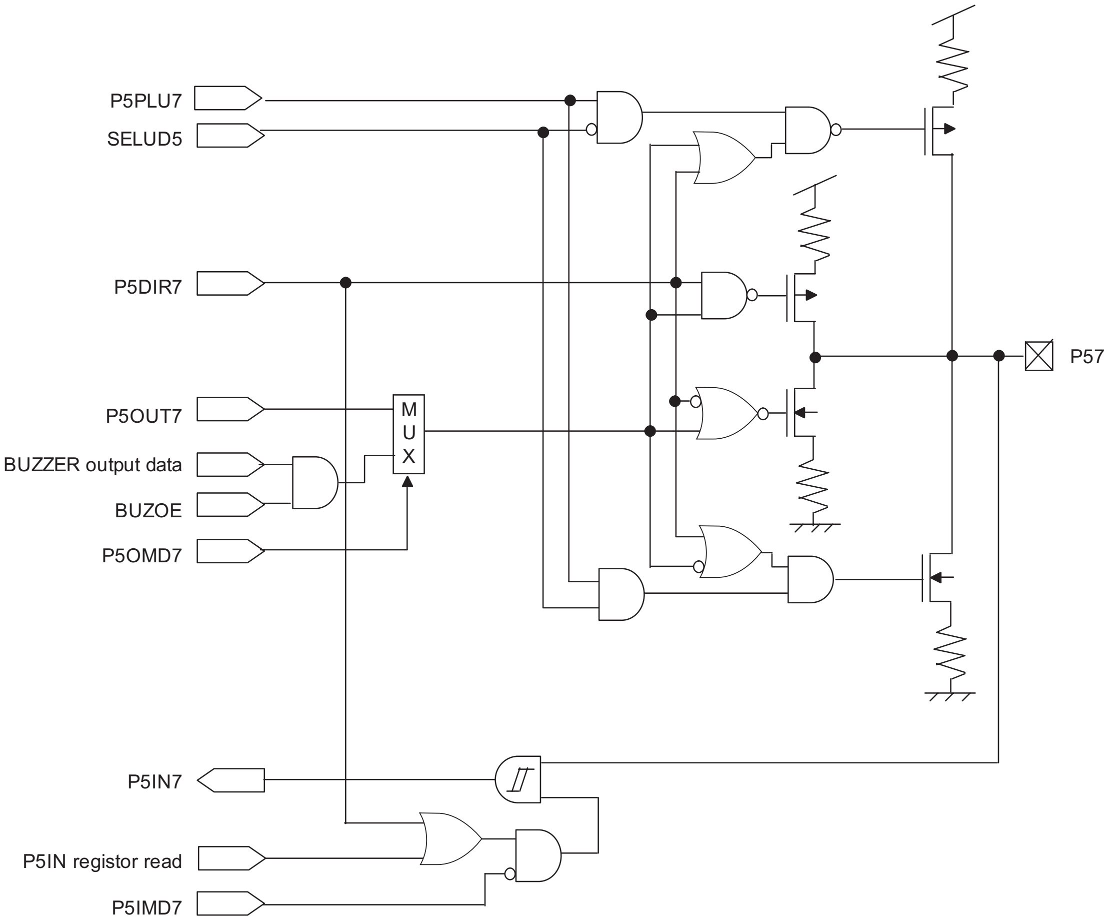


Figure:5.7.18 P57 Block Diagram (MN101EFA2)

5.8 Port 6

5.8.1 Description

■ General Pin Setup

To output data to pin, set the control flag of P6DIR register to "1" and write data to P6OUT register.
To read input data of pin, set the control flag of P6DIR register to "0" and read the value of P6IN register.

Each bit can be set individually to either an input or output by P6DIR register. The control flag of P6DIR register is set to "1" for output mode,

Each bit can be set individually if pull-up resistor is added or not, by P6PLU register. Set the control flag of P6PLU register to "1" to add pull-up resistor.

Each bit can be selected individually as Nch open-drain output by P6ODC register. The control flag of P6ODC register is set to "1" for Nch open-drain output, and "0" for push-pull output.

■ Special Function Pin Setup

P62 is also used as the I/O pin of timer 1.
P63 is also used as the I/O pin of timer 3.

Each bit for the I/O mode can be set individually by P6OMD registers.
These registers are set to "1" to output the special function data, and "0" to be used as the general port.

P65 is also used as the I/O pin of serial 2 transmission/reception data and UART2 transmission data. When the SC2SBOS flag of SC2MD1 register is "1", P65 is the serial data I/O pin. Push-pull output or Nch open-drain output can be selected by setting P6ODC register.

P66 is also used as the input pin of serial 2 reception data and UART2 reception data. When the SC2SBIS flag of SC2MD1 register is "1", P65 is the serial data input pin.

P67 is also used as the I/O pin of serial 2 clock. When the SC2SBTS flag of the serial interface 2 mode register 1 (SC2MD1) is "1", P67 is the I/O pin of the serial clock. Push-pull output or Nch open-drain output can be selected by setting P6ODC register.

Table:5.8.1 The port 6 Special Function Pins

Table remarks √: With function -: Without function

Pins	Special Functions	MN101EFA8	MN101EFA3	MN101EFA7	MN101EFA2
P62	TM1IOB	√	√	√	√
	TS0OP	-	-	√	√
P63	TM3IOB	√	√	√	√
P64	-	√	√	√	√
P65	SBO2	√	√	√	√
	TXD2	√	√	√	√
P66	SBI2	√	√	√	√
	RXD2	√	√	√	√
P67	SBT2	√	√	√	√

5.8.2 Registers

■ Port 6 Output Register (P6OUT: 0x03E76)

bp	7	6	5	4	3	2	1	0
Flag	P6OUT7	P6OUT6	P6OUT5	P6OUT4	P6OUT3	P6OUT2	-	-
At reset	X	X	X	X	X	X	-	-
Access	R/W	R/W	R/W	R/W	R/W	R/W	-	-

bp	Flag	Description
7-2	P6OUT7-2	Output data 0: Output "Low" (V_{SS} level) 1: Output "High" (V_{DD5} level)
1-0	-	-

■ Port 6 Input Register (P6IN: 0x03E86)

bp	7	6	5	4	3	2	1	0
Flag	P6IN7	P6IN6	P6IN5	P6IN4	P6IN3	P6IN2	-	-
At reset	X	X	X	X	X	X	-	-
Access	R	R	R	R	R	R	-	-

bp	Flag	Description
7-2	P6IN7-2	Input data 0: Pin is "Low" (V_{SS} level) 1: Pin is "High" (V_{DD5} level)
1-0	-	-

■ Port 6 Direction Control Register (P6DIR: 0x03E96)

bp	7	6	5	4	3	2	1	0
Flag	P6DIR7	P6DIR6	P6DIR5	P6DIR4	P6DIR3	P6DIR2	-	-
At reset	0	0	0	0	0	0	-	-
Access	R/W	R/W	R/W	R/W	R/W	R/W	-	-

bp	Flag	Description
7-2	P6DIR7-2	I/O mode selection 0: Input mode 1: Output mode
1-0	-	-

- Port 6 Pull-up Resistor Control Register (P6PLU: 0x03EA6)

bp	7	6	5	4	3	2	1	0
Flag	P6PLU7	P6PLU6	P6PLU5	P6PLU4	P6PLU3	P6PLU2	-	-
At reset	0	0	0	0	0	0	-	-
Access	R/W	R/W	R/W	R/W	R/W	R/W	-	-

bp	Flag	Description
7-0	P6PLU7-0	Pull-up resistor selection 0: Not added 1: Added

- Port 6 Output Mode Register (P6OMD: 0x03EB6)

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	P6OMD3	P6OMD2	-	-
At reset	-	-	-	-	0	0	-	-
Access	-	-	-	-	R/W	R/W	-	-

bp	Flag	Description
7-4	-	-
3	P6OMD3	I/O port or special function selection 0: P63 1: TM3IOB
2	P6OMD2	I/O port or special function selection 0: P62 1: TM1IOB
1-0	-	-

■ Port 6 Nch Open-drain Control Register (P6ODC: 0x03EF6)

bp	7	6	5	4	3	2	1	0
Flag	P6ODC7	-	P6ODC5	-	-	-	-	-
At reset	0	-	0	-	-	-	-	-
Access	R/W	-	R/W	-	-	-	-	-

bp	Flag	Description
7	P6ODC7	P67 Nch open-drain output selection 0: Push-pull output 1: Nch open-drain output
6	-	-
5	P6ODC5	P65 Nch open-drain output selection 0: Push-pull output 1: Nch open-drain output
4-0	-	-

5.8.3 Block Diagram

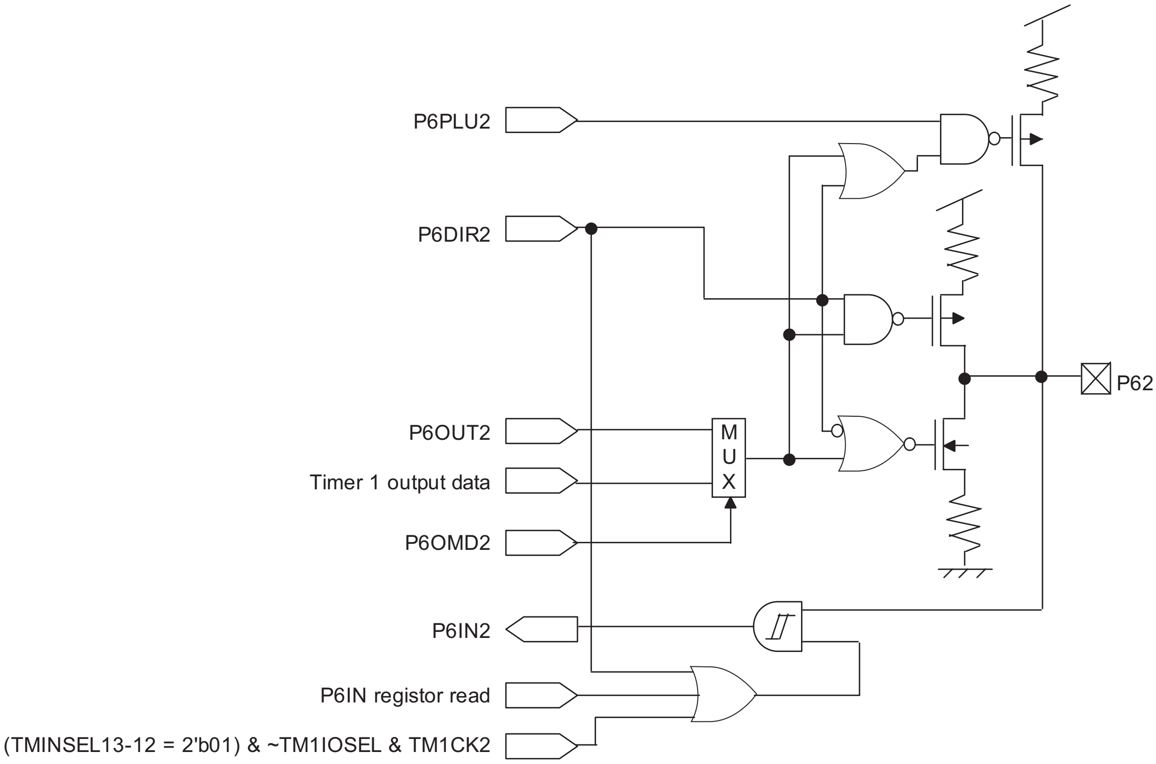


Figure:5.8.1 P62 Block Diagram (MN101EFA8/A3/A2)

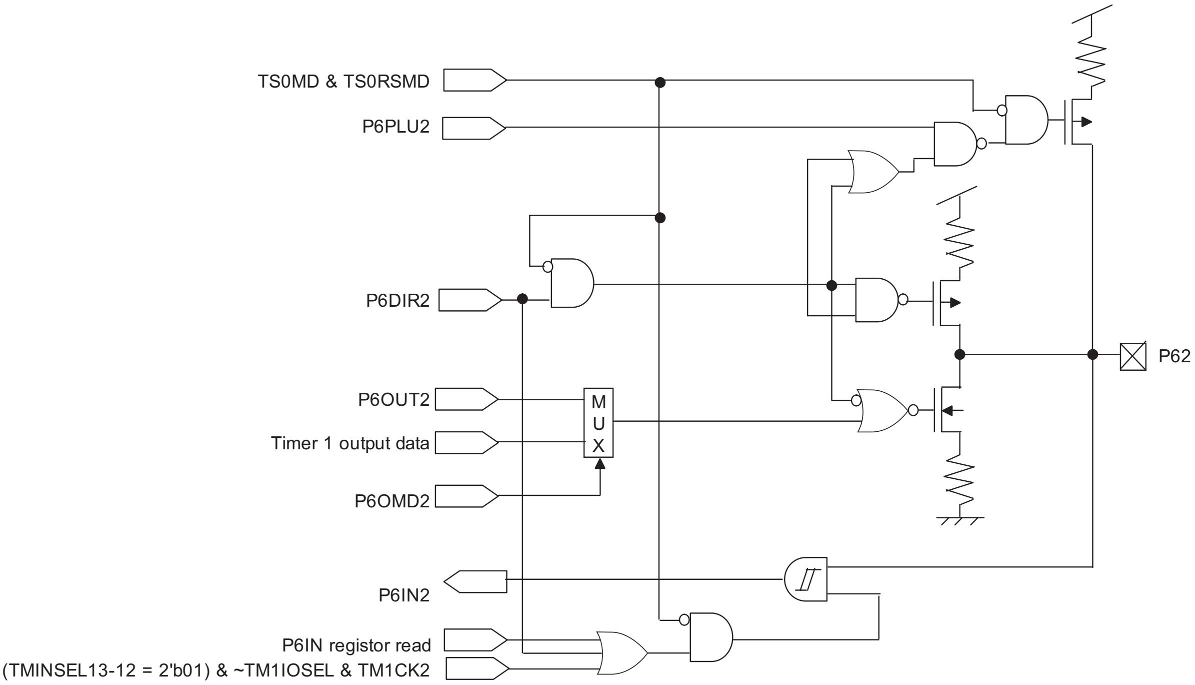


Figure:5.8.2 P62 Block Diagram (MN101EFA7)

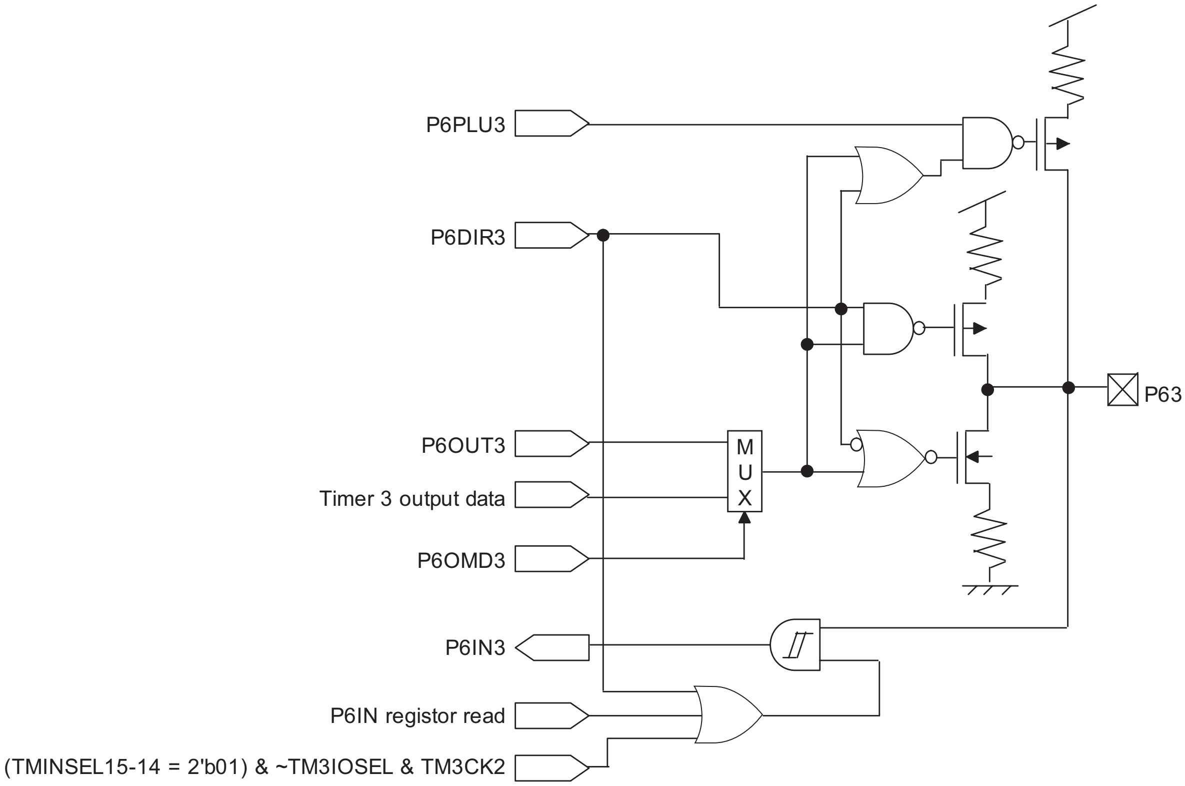


Figure:5.8.3 P63 Block Diagram (MN101EFA8/A7/A3/A2)

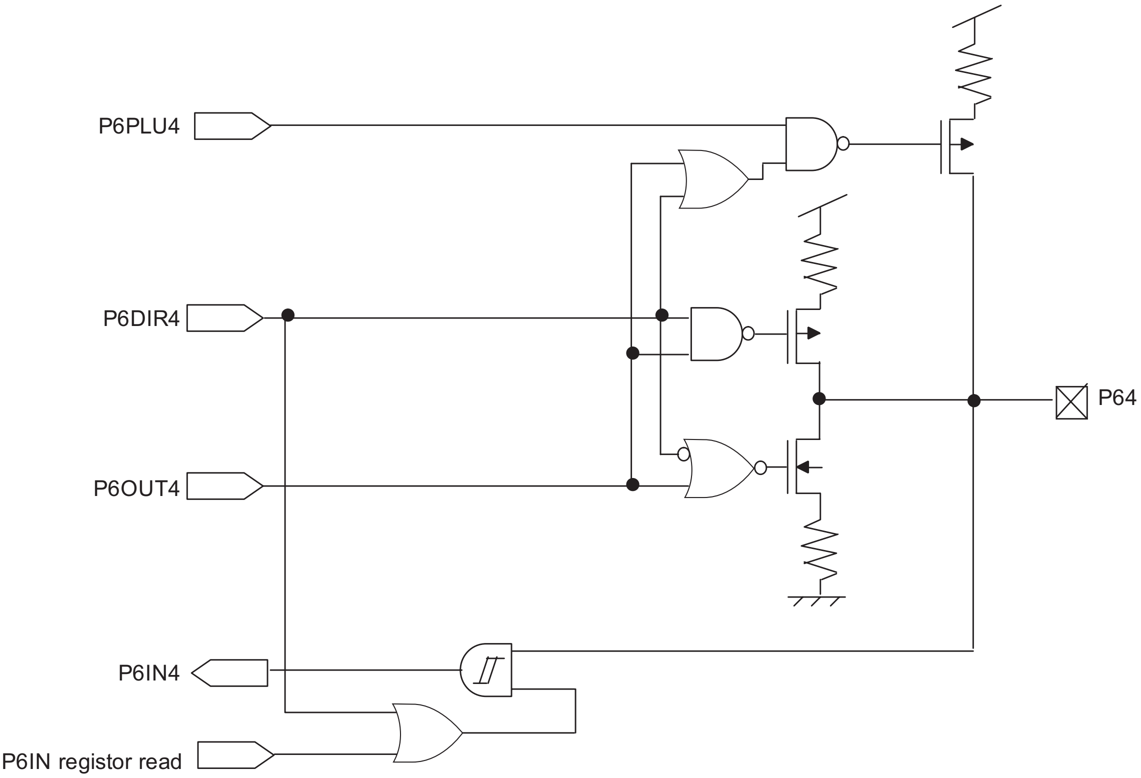


Figure:5.8.4 P64 Block Diagram (MN101EFA8/A7/A3/A2)

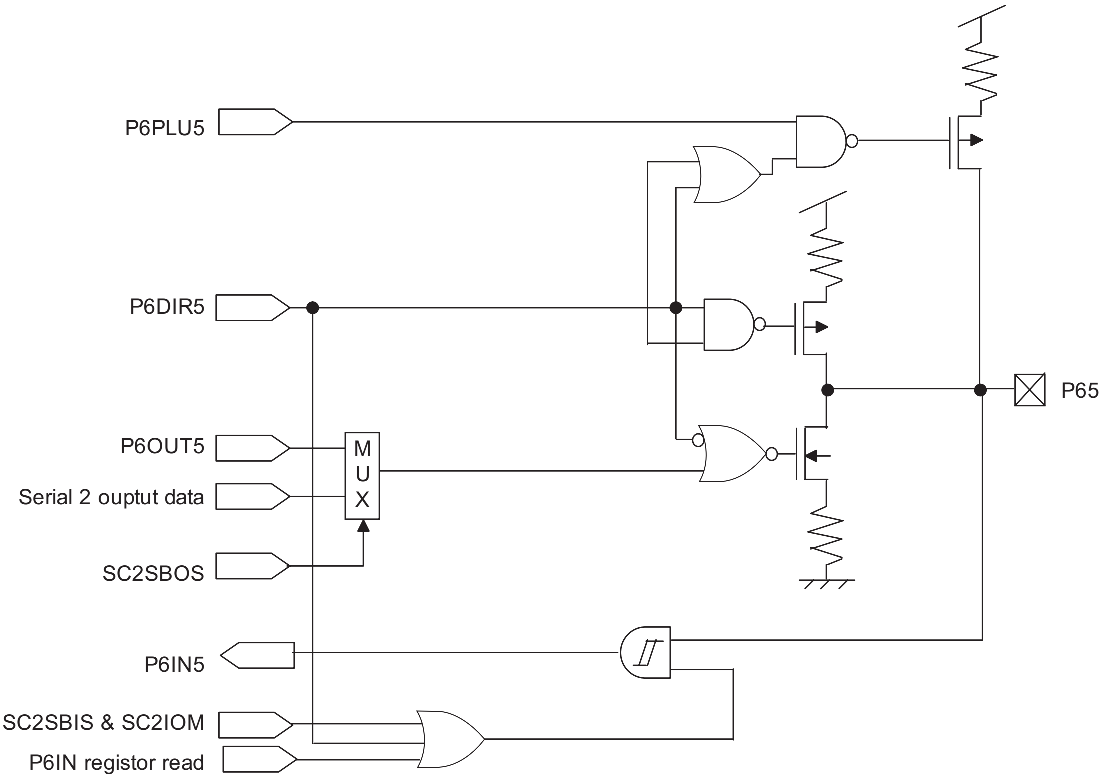


Figure:5.8.5 P65 Block Diagram (MN101EFA8/A7/A3/A2)

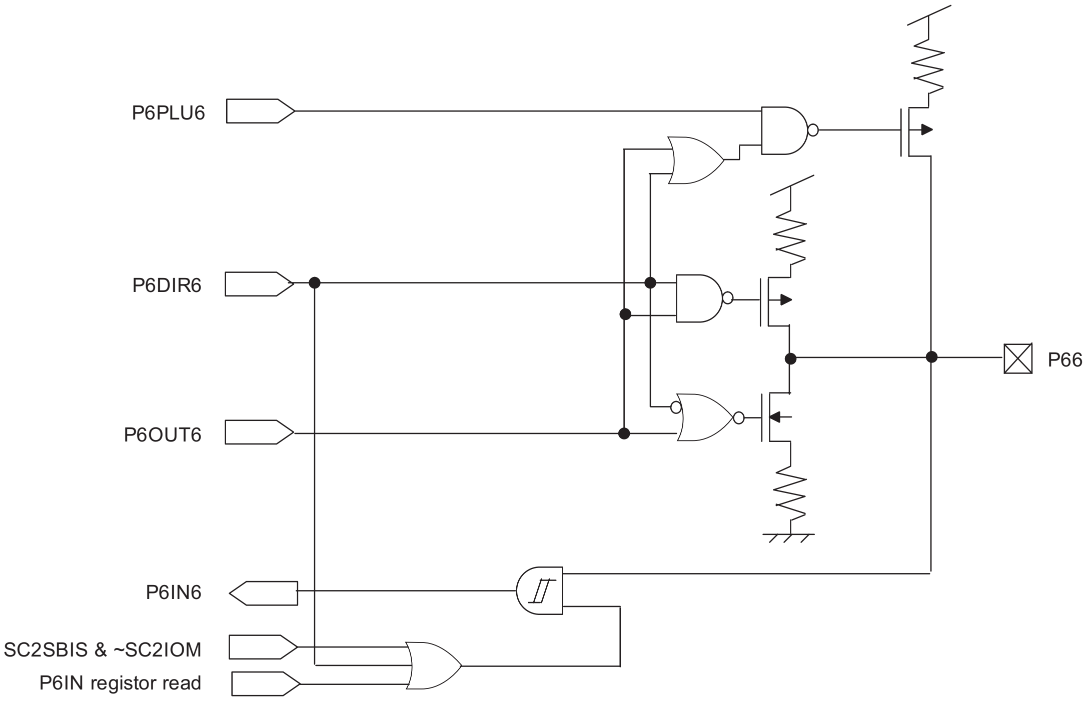


Figure:5.8.6 P66 Block Diagram (MN101EFA8/A7/A3/A2)

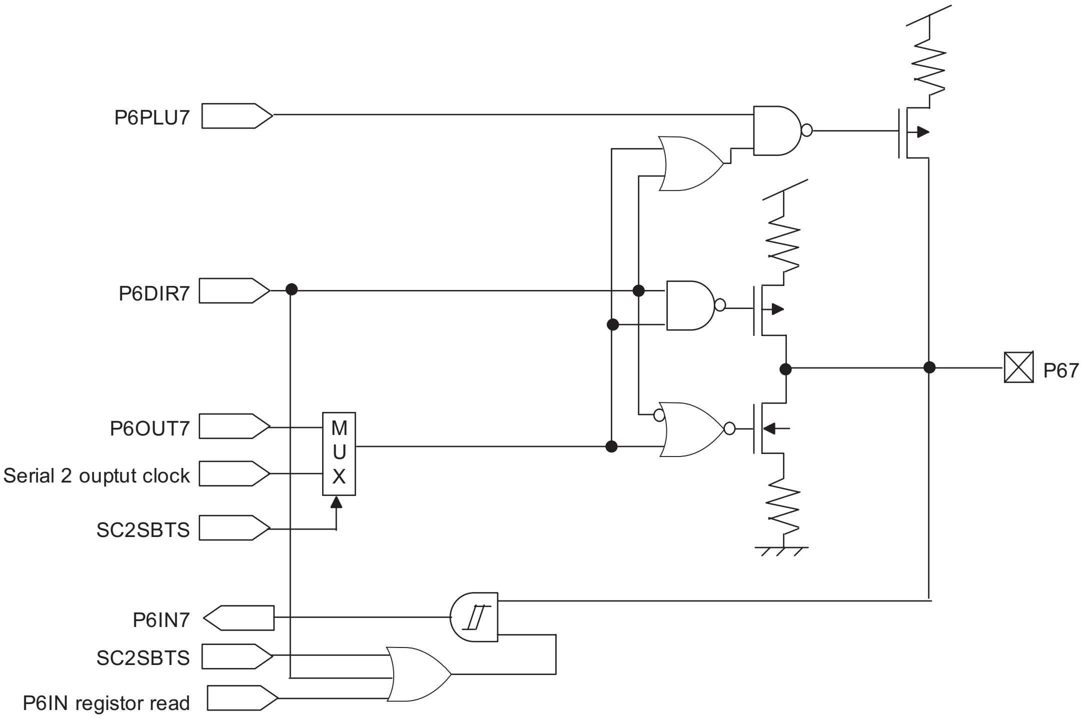


Figure:5.8.7 P67 Block Diagram (MN101EFA8/A7/A3/A2)

5.9 Port 7

5.9.1 Description

■ General Port Setup

To output data to pin, set the control flag of P7DIR register to "1" and write data to P7OUT register.
To read input data of pin, set the control flag of P7DIR register to "0" and read the value of P7IN register.

Each bit can be set individually to either an input or output by P7DIR register. The control flag of P7DIR register is set to "1" for output mode, and "0" for input mode.

Each bit can be set individually if pull-up resistor is added or not, by P7PLU register. Set the control flag of P7PLU register to "1" to add pull-up resistor.

Each bit can be selected individually as Nch open-drain output by P7ODC register. The control flag of P7ODC register is set to "1" for Nch open-drain output, and "0" for push-pull output.

■ Special Function Pin Setup

P70 to P77 are also used as the input pin of KEY interrupt.

P70 is also used as the input pin of the serial 4 reception data. When the SC4SBIS flag of serial 4 mode register 1 (SC4MD1) is "1", P70 is the input pin of serial data.

P71 is also used as the I/O pin of serial 4 transmission/reception data and IIC4 transmission/reception data. When the SC4SBOS flag of SC4MD1 register is "1", P71 is the I/O pin of the serial data. Push-pull output or Nch open-drain output can be selected by setting P7ODC register.

P72 is also used as the I/O pin of serial 4 clock and the output pin of IIC4 clock. When the SC4SBTS flag of SC4MD1 register is "1", P72 is the serial clock I/O pin. Push-pull output or Nch open-drain output can be selected by setting P7ODC register.

P75 is also used as the I/O pin of serial 1 transmission/reception data and UART1 transmission data. When the SC1SBOS flag of SC1MD1 register is "1", P75 is the serial data I/O pin. Push-pull output or Nch open-drain output can be selected by setting P7ODC register.

P76 is also used as the input pin of serial 1 reception data and UART1 reception data. When the SC1SBIS flag of SC1MD1 register is "1", P76 is the serial data input pin.

P77 is also used as the I/O pin of serial 1 clock. When the SC1SBTS flag of the serial interface 1 mode register 1 (SC1MD1) is "1", P77 is the I/O pin of the serial clock. Push-pull output or Nch open-drain output can be selected by setting P7ODC register.

Table:5.9.1 The port 7 Special Function Pins

Table remarks √: With function -: Without function

Pins	Special Functions	MN101EFA8	MN101EFA3	MN101EFA7	MN101EFA2
P70	KEY0	√	√	√	√
	SBI4A	√	√	√	√
P71	KEY1	√	√	√	√
	SBO4A	√	√	√	√
	SDA4A	√	√	√	√
P72	KEY2	√	√	√	√
	SBT4A	√	√	√	√
	SCL4A	√	√	√	√
P73	KEY3	√	√	√	√
P74	KEY4	√	√	√	√
P75	KEY5	√	√	√	√
	SBO1B	√	√	√	√
	TXD1B	√	√	√	√
P76	KEY6	√	√	√	√
	SBI1B	√	√	√	√
	RXD1B	√	√	√	√
P77	KEY7	√	√	√	√
	SBT1B	√	√	√	√

5.9.2 Registers

■ Port 7 Output Register (P7OUT: 0x03E77)

bp	7	6	5	4	3	2	1	0
Flag	P7OUT7	P7OUT6	P7OUT5	P7OUT4	P7OUT3	P7OUT2	P7OUT1	P7OUT0
At reset	X	X	X	X	X	X	X	X
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-0	P7OUT7-0	Output data 0: Output "Low" (V_{SS} level) 1: Output "High" (V_{DD5} level)

■ Port 7 Input Register (P7IN: 0x03E87)

bp	7	6	5	4	3	2	1	0
Flag	P7IN7	P7IN6	P7IN5	P7IN4	P7IN3	P7IN2	P7IN1	P7IN0
At reset	X	X	X	X	X	X	X	X
Access	R	R	R	R	R	R	R	R

bp	Flag	Description
7-0	P7IN7-0	Input data 0: Pin is "Low" (V_{SS} level) 1: Pin is "High" (V_{DD5} level)

■ Port 7 Direction Control Register (P7DIR: 0x03E97)

bp	7	6	5	4	3	2	1	0
Flag	P7DIR7	P7DIR6	P7DIR5	P7DIR4	P7DIR3	P7DIR2	P7DIR1	P7DIR0
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-0	P7DIR7-0	I/O mode selection 0: Input mode 1: Output mode

- Port 7 Pull-up Resistor Control Register (P7PLU: 0x03EA7)

bp	7	6	5	4	3	2	1	0
Flag	P7PLU7	P7PLU6	P7PLU5	P7PLU4	P7PLU3	P7PLU2	P7PLU1	P7PLU0
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-0	P7PLU7-0	Pull-up resistor selection 0: Not added 1: Added

- Port 7 Nch Open-drain Control Register (P7ODC: 0x03EF7)

bp	7	6	5	4	3	2	1	0
Flag	P7ODC7	-	P7ODC5	-	-	P7ODC2	P7ODC1	-
At reset	0	-	0	-	-	0	0	-
Access	R/W	-	R/W	-	-	R/W	R/W	-

bp	Flag	Description
7	P7ODC7	P77 Nch open-drain output selection 0: Push-pull output 1: Nch open-drain output
6	-	-
5	P7ODC5	P75 Nch open-drain output selection 0: Push-pull output 1: Nch open-drain output
4-3	-	-
2	P7ODC2	P72 Nch open-drain output selection 0: Push-pull output 1: Nch open-drain output
1	P7ODC1	P71 Nch open-drain output selection 0: Push-pull output 1: Nch open-drain output
0	-	-

5.9.3 Block Diagram

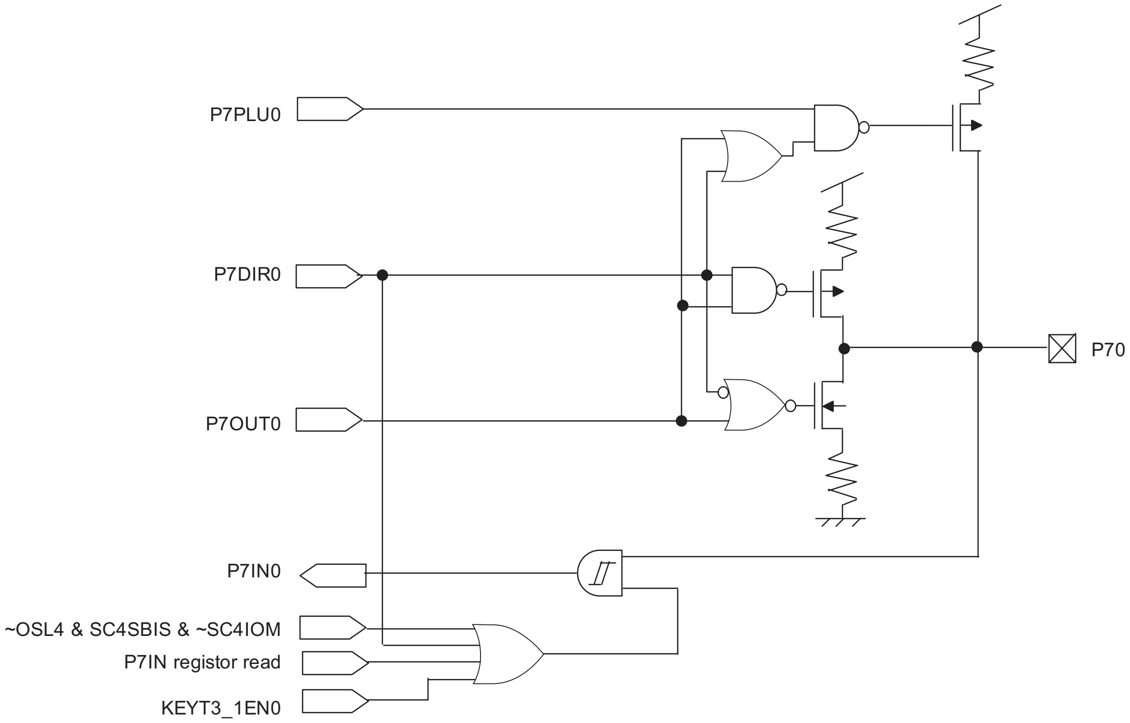


Figure:5.9.1 P70 Block Diagram (MN101EFA8/A7/A3/A2)

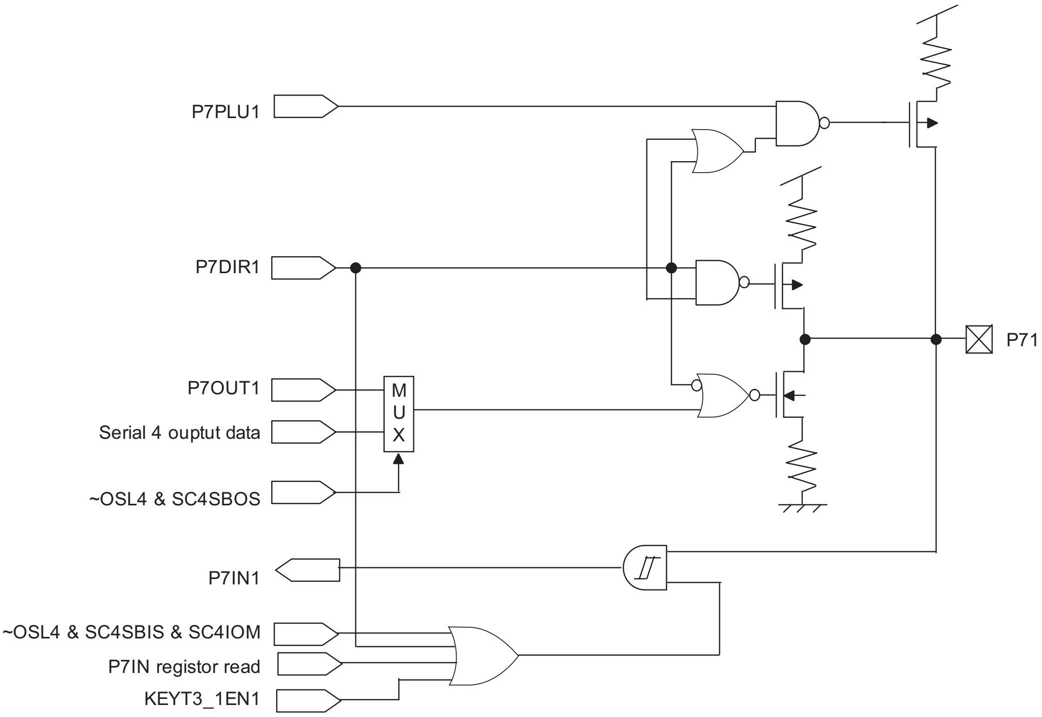


Figure:5.9.2 P71 Block Diagram (MN101EFA8/A7/A3/A2)

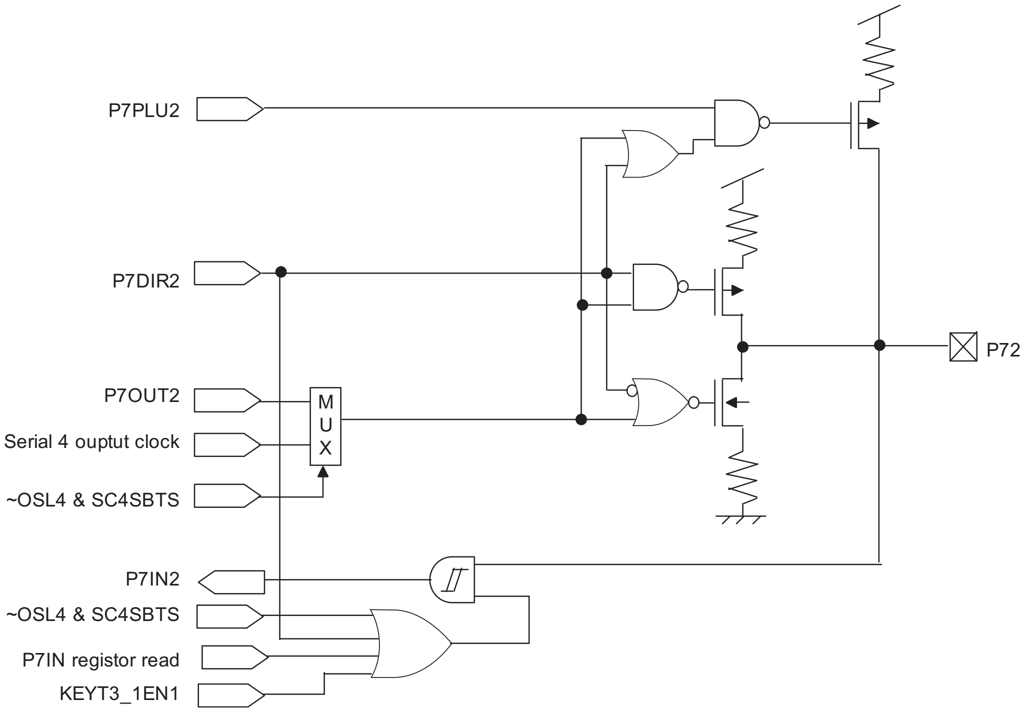


Figure:5.9.3 P72 Block Diagram (MN101EFA8/A7/A3/A2)

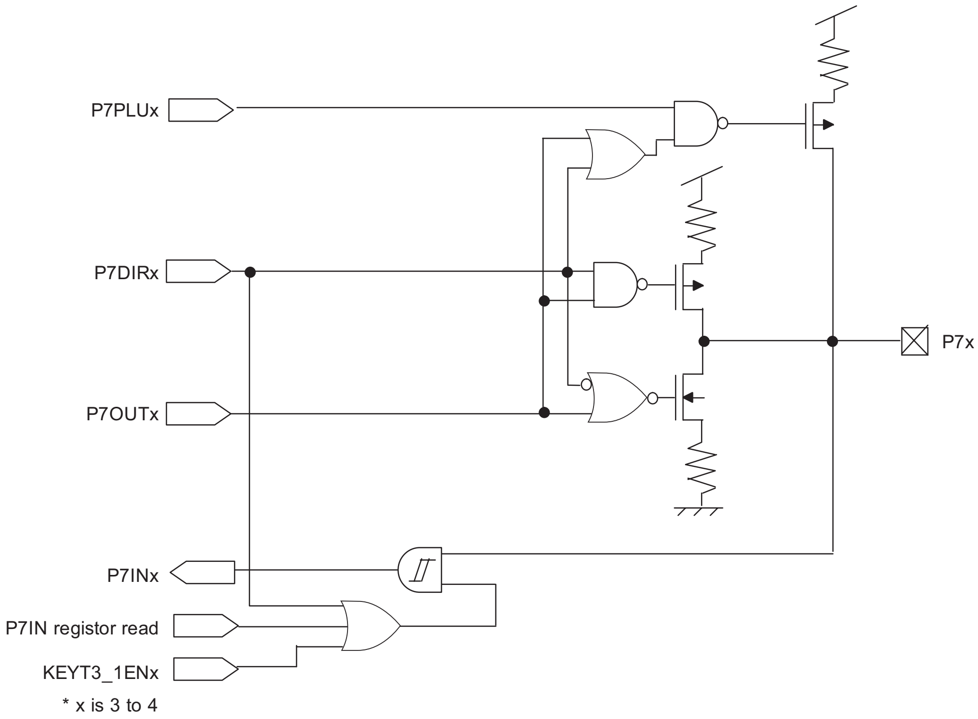


Figure:5.9.4 P7x Block Diagram (MN101EFA8/A7/A3/A2)

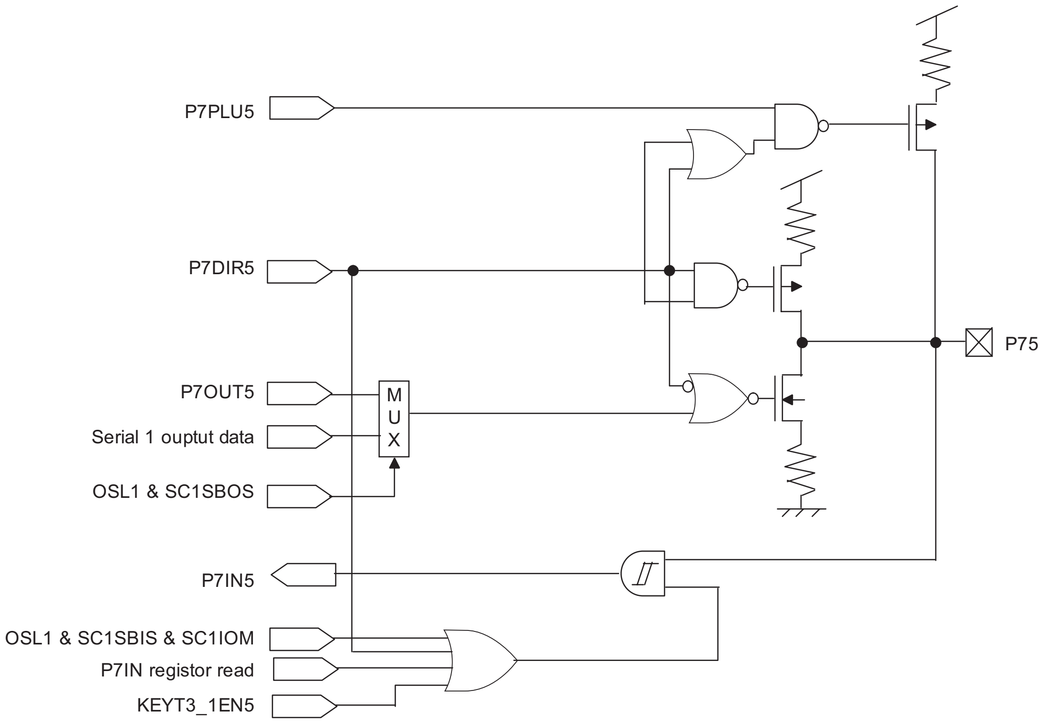


Figure:5.9.5 P75 Block Diagram (MN101EFA8/A7/A3/A2)

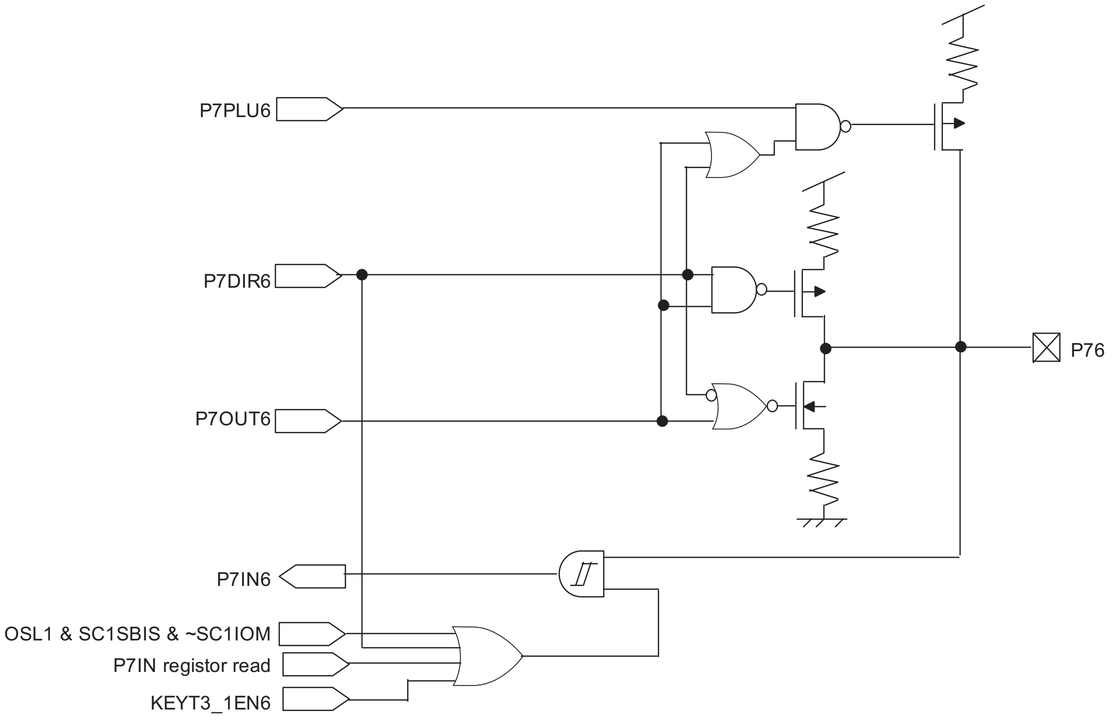


Figure:5.9.6 P76 Block Diagram (MN101EFA8/A7/A3/A2)

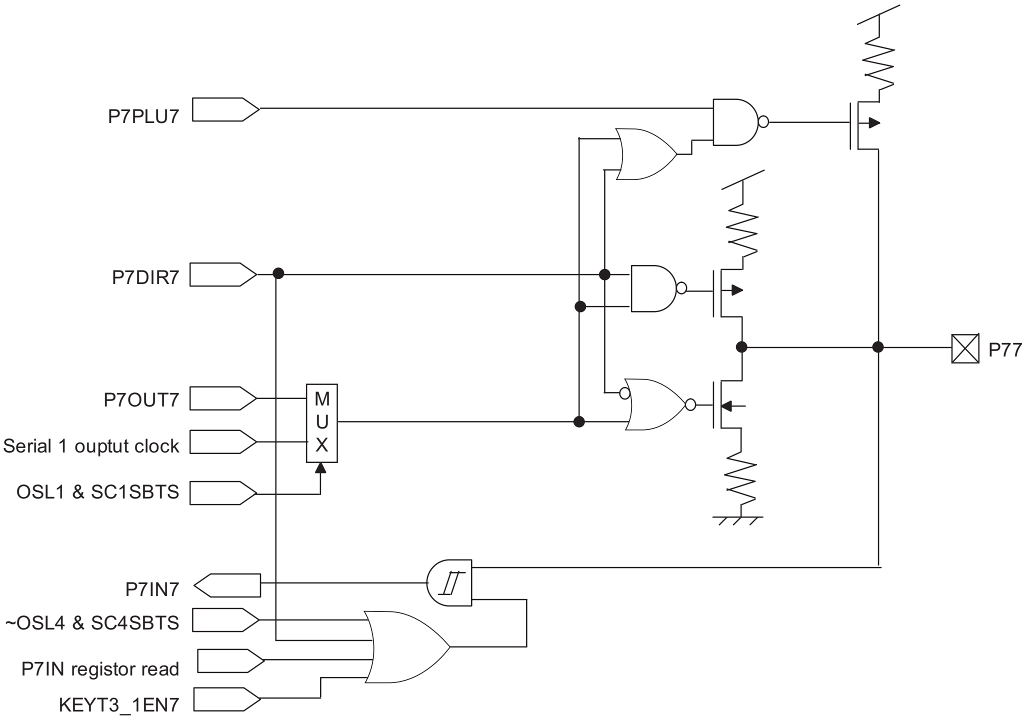


Figure:5.9.7 P77 Block Diagram (MN101EFA8/A7/A3/A2)

5.10 Port 8

5.10.1 Description

■ General Port Setup

To output data to pin, set the control flag of P8DIR register to "1" and write data to P8OUT register.
To read input data of pin, set the control flag of P8DIR register to "0" and read the value of P8IN register.

Each bit can be set individually to either an input or output by P8DIR register. The control flag of P8DIR register is set to "1" for output mode, and "0" for input mode.

Each bit can be set individually if pull-up resistor is added or not, by P8PLU register. Set the control flag of P8PLU register to "1" to add pull-up resistor.

■ Special Function Pin Setup

P80 to P85 are also used as the I/O pin of timer 9 PWM output pin.

P86 is also used as the I/O pin of reverse buzzer.
P87 is also used as the I/O pin of buzzer.

Each bit for the I/O mode can be set individually by P8OMD registers.
These registers are set to "1" to output the special function data, and "0" to be used as the general port.

Table:5.10.1 The port 8 Special Function Pins

Table remarks √: With function -: Without function

Pins	Special Functions	MN101EFA8	MN101EFA3	MN101EFA7	MN101EFA2
P80	TM9OD0	√	√	√	√
P81	TM9OD1	√	√	√	√
P82	TM9OD2	√	√	√	√
P83	TM9OD3	√	√	√	√
P84	TM9OD4	√	√	√	√
P85	TM9OD5	√	√	√	√
P86	NBUZZERB	√	√	√	√
P87	BUZZERB	√	√	√	√

5.10.2 Registers

■ Port 8 Output Register (P8OUT: 0x03E78)

bp	7	6	5	4	3	2	1	0
Flag	P8OUT7	P8OUT6	P8OUT5	P8OUT4	P8OUT3	P8OUT2	P8OUT1	P8OUT0
At reset	X	X	X	X	X	X	X	X
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-0	P8OUT7-0	Output data 0: Output "Low" (V_{SS} level) 1: Output "High" (V_{DD5} level)

■ Port 8 Input Register (P8IN: 0x03E88)

bp	7	6	5	4	3	2	1	0
Flag	P8IN7	P8IN6	P8IN5	P8IN4	P8IN3	P8IN2	P8IN1	P8IN0
At reset	X	X	X	X	X	X	X	X
Access	R	R	R	R	R	R	R	R

bp	Flag	Description
7-0	P8IN7-0	Input data 0: Pin is "Low" (V_{SS} level) 1: Pin is "High" (V_{DD5} level)

■ Port 8 Direction Control Register (P8DIR: 0x03E98)

bp	7	6	5	4	3	2	1	0
Flag	P8DIR7	P8DIR6	P8DIR5	P8DIR4	P8DIR3	P8DIR2	P8DIR1	P8DIR0
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-0	P8DIR7-0	I/O mode selection 0: Input mode 1: Output mode

- Port 8 Pull-up Resistor Control Register (P8PLU: 0x03EA8)

bp	7	6	5	4	3	2	1	0
Flag	P8PLU7	P8PLU6	P8PLU5	P8PLU4	P8PLU3	P8PLU2	P8PLU1	P8PLU0
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-0	P8PLU7-0	Pull-up resistor selection 0: Not added 1: Added

- Port 8 Output Mode Register (P8OMD: 0x03EB8)

bp	7	6	5	4	3	2	1	0
Flag	P8OMD7	P8OMD6	P8OMD5	P8OMD4	P8OMD3	P8OMD2	P8OMD1	P8OMD0
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7	P8OMD7	I/O port or special function selection 0: P87 1: BUZZERB
6	P8OMD6	I/O port or special function selection 0: P86 1: NBUZZERB
5	P8OMD5	I/O port or special function selection 0: P85 1: TM9OD5
4	P8OMD4	I/O port or special function selection 0: P84 1: TM9OD4
3	P8OMD3	I/O port or special function selection 0: P83 1: TM9OD3
2	P8OMD2	I/O port or special function selection 0: P82 1: TM9OD2
1	P8OMD1	I/O port or special function selection 0: P81 1: TM9OD1
0	P8OMD0	I/O port or special function selection 0: P80 1: TM9OD0

5.10.3 Block Diagram

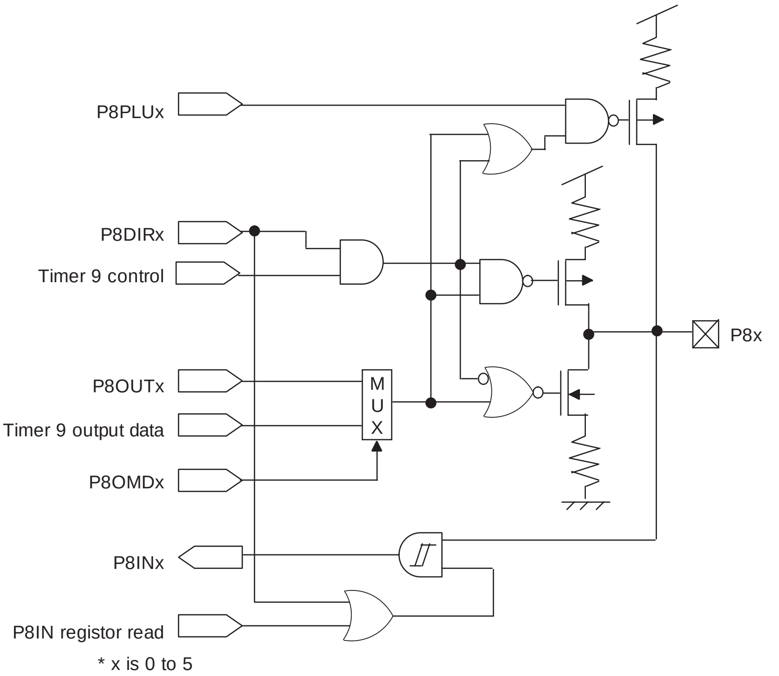


Figure:5.10.1 P8x Block Diagram (MN101EFA8/A7/A3/A2)

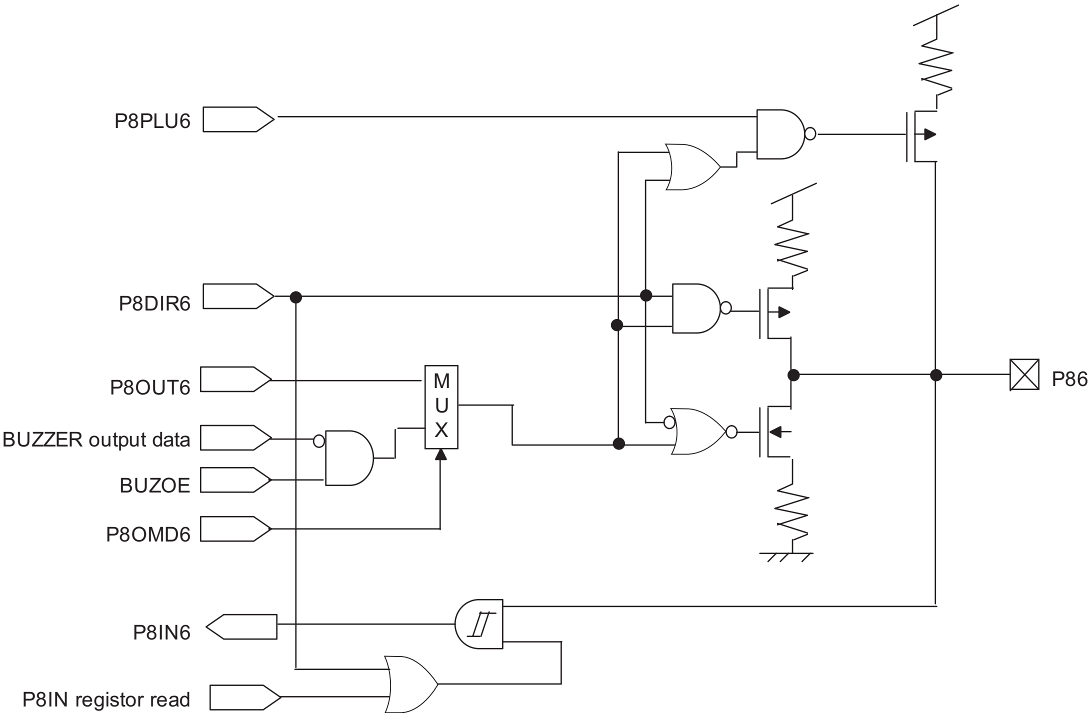


Figure:5.10.2 P86 Block Diagram (MN101EFA8/A7/A3/A2)

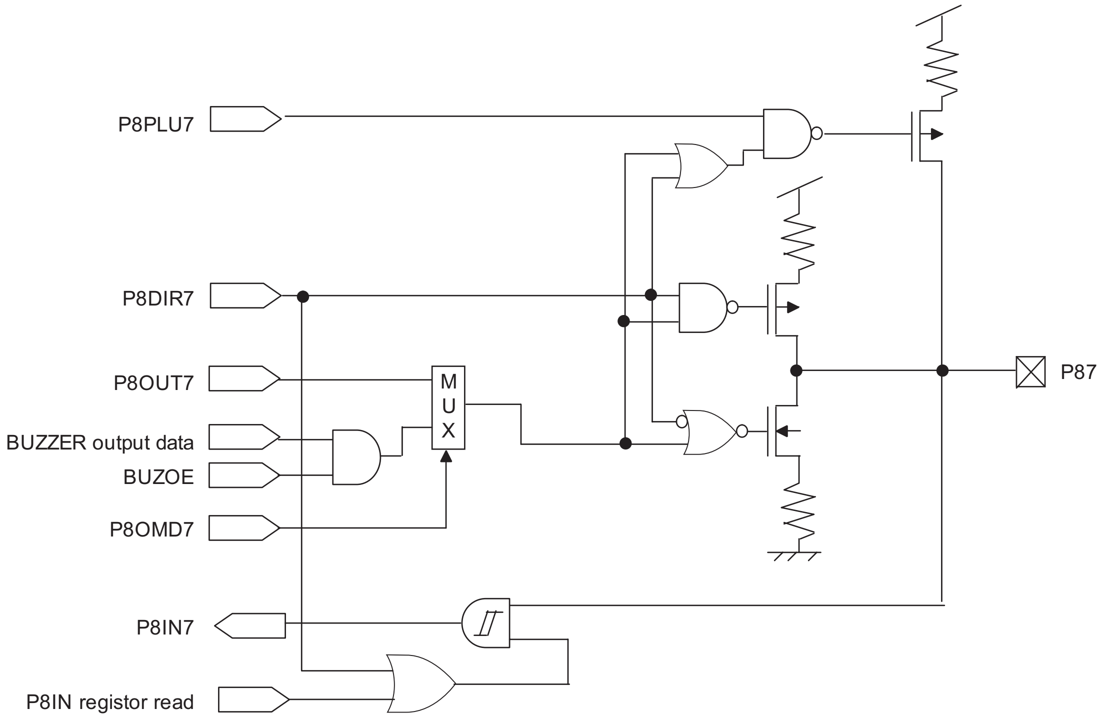


Figure:5.10.3 P87 Block Diagram (MN101EFA8/A7/A3/A2)

5.11 Port 9

5.11.1 Description

MN101EFA7/A2 do not include P92 to P93.

■ General Port Setup

To output data to pin, set the control flag of P9DIR register to "1" and write data to P9OUT register.
To read input data of pin, set the control flag of P9DIR register to "0" and read the value of P9IN register.

Each bit can be set individually to either an input or output by P9DIR register. The control flag of P9DIR register is set to "1" for output mode, and "0" for input mode.

Each bit can be set individually if pull-up resistor is added or not by P9PLUD register. Set the control flag of P9PLUD register to "1" to add pull-up or pull-down resistor.
Port 9 can be selected to add pull-up or pull-down register by the SELUD9 flag of SELUD2 register.

Each bit can be selected individually as input mode by P9IMD register. The control flag of P9IMD register is set to "1" to input the special function data and the value read from P9IN register is read to undefined, and "0" to use as the general port.

■ Special Function Pin Setup

P90 to P91 is also used as low-speed oscillator. When the XI/XO selection flag of the external low-speed oscillation control register (OSCSCNT) is "1", P90 to P91 can be used as low-speed oscillator.

P92 to P94 are also used as analog input pin. Input mode for each bit can be selected by P9IMD register. When these pins are used as analog input pin, the value read from the port 9 input register is read to be "0".

(This function is equipped in MN101EFA7/A2.)
P94 is also used as analog input pin. Input mode for each bit can be selected by P9IMD register.
When P94 is used as analog input pin, the value read from the port 9 input register is read to be "0".

(This function is equipped in MN101EFA8.)
P92 to P94 are also used as input pins for touch sensor timer.
Set "Used" to corresponding channel by TS0TCHSEL register. Refer to[Chapter XV Touch Sensor Timer].

(This function is equipped in MN101EFA7.)
P94 is also used as input pins for touch sensor timer.
Set "Used" to corresponding channel by TS0TCHSEL register. Refer to [Chapter XV Touch Sensor Timer].

Table:5.11.1 The port 9 Special Function Pins

Table remarks √: With function -: Without function

Pins	Special Functions	MN101EFA8	MN101EFA3	MN101EFA7	MN101EFA2
P90	XI	√	√	√	√
P91	XO	√	√	√	√
P92	AN14	√	√	-	-
	TSIN06	√	-	-	-
P93	AN13	√	√	-	-
	TSIN05	√	-	-	-
P94	AN12	√	√	-	-
	TSIN04	√	-	-	-
	AN8	-	-	√	√
	TSIN00	-	-	√	-

5.11.2 Registers

■ Port 9 Output Register (P9OUT: 0x03E79)

MN101EFA8/MN101EFA3

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	P9OUT4	P9OUT3	P9OUT2	P9OUT1	P9OUT0
At reset	-	-	-	X	X	X	X	X
Access	-	-	-	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-5	-	-
4-0	P9OUT4-0	Output data 0: Output "Low" (V_{SS} level) 1: Output "High" (V_{DD5} level)

■ Port 9 Output Register (P9OUT: 0x03E79)

MN101EFA7/MN101EFA2

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	P9OUT4	-	-	P9OUT1	P9OUT0
At reset	-	-	-	X	-	-	X	X
Access	-	-	-	R/W	-	-	R/W	R/W

bp	Flag	Description
7-5	-	-
4	P9OUT4	Output data 0: Output "Low" (V_{SS} level) 1: Output "High" (V_{DD5} level)
3-2	-	-
1-0	P9OUT6-0	Output data 0: Output "Low" (V_{SS} level) 1: Output "High" (V_{DD5} level)

■ Port 9 Input Register (P9IN: 0x03E89)

MN101EFA8/MN101EFA3

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	P9IN4	P9IN3	P9IN2	P9IN1	P9IN0
At reset	-	-	-	X	X	X	X	X
Access	-	-	-	R	R	R	R	R

bp	Flag	Description
7-5	-	-
4-0	P9IN4-0	Input data 0: Pin is "Low" (V_{SS} level) 1: Pin is "High" (V_{DD5} level)

■ Port 9 Input Register (P9IN: 0x03E89)

MN101EFA7/MN101EFA2

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	P9IN4	-	-	P9IN1	P9IN0
At reset	-	-	-	X	-	-	X	X
Access	-	-	-	R	-	-	R	R

bp	Flag	Description
7-5	-	-
4	P9IN4	Input data 0: Pin is "Low" (V_{SS} level) 1: Pin is "High" (V_{DD5} level)
3-2	-	-
1-0	P9IN1-0	Input data 0: Pin is "Low" (V_{SS} level) 1: Pin is "High" (V_{DD5} level)

■ Port 9 Direction Control Register (P9DIR: 0x03E99)

MN101EFA8/MN101EFA3

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	P9DIR4	P9DIR3	P9DIR2	P9DIR1	P9DIR0
At reset	-	-	-	0	0	0	0	0
Access	-	-	-	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-5	-	-
4-0	P9DIR4-0	I/O mode selection 0: Input mode 1: Output mode

■ Port 9 Direction Control Register (P9DIR: 0x03E99)

MN101EFA7/MN101EFA2

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	P9DIR4	-	-	P9DIR1	P9DIR0
At reset	-	-	-	0	-	-	0	0
Access	-	-	-	R/W	-	-	R/W	R/W

bp	Flag	Description
7-5	-	-
4	P9DIR4	I/O mode selection 0: Input mode 1: Output mode
3-2	-	-
1-0	P9DIR1-0	I/O mode selection 0: Input mode 1: Output mode

■ Port 9 Pull-up/pull-down Resistor Control Register (P9PLUD: 0x03EA9)

MN101EFA8/MN101EFA3

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	P9PLUD4	P9PLUD3	P9PLUD2	P9PLUD1	P9PLUD0
At reset	-	-	-	0	0	0	0	0
Access	-	-	-	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-5	-	-
4-0	P9PLUD4-0	Pull-up/pull-down resistor selection 0: Not added 1: Added

■ Port 9 Pull-up/pull-down Resistor Control Register (P9PLUD: 0x03EA9)

MN101EFA7/MN101EFA2

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	P9PLUD4	-	-	P9PLUD1	P9PLUD0
At reset	-	-	-	0	-	-	0	0
Access	-	-	-	R/W	-	-	R/W	R/W

bp	Flag	Description
7-5	-	-
4	P9PLUD4	Pull-up/pull-down resistor selection 0: Not added 1: Added
3-2	-	-
1-0	P9PLUD1-0	Pull-up/pull-down resistor selection 0: Not added 1: Added

■ Port 9 Input Mode Register 1 (P9IMD: 0x03EC9)

MN101EFA8/MN101EFA3

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	P9IMD4	P9IMD3	P9IMD2	-	-
At reset	-	-	-	0	0	0	-	-
Access	-	-	-	R/W	R/W	R/W	-	-

bp	Flag	Description
7-5	-	-
4	P9IMD4	I/O port or analog input selection 0: P94 1: AN12
3	P9IMD3	I/O port or analog input selection 0: P93 1: AN13
2	P9IMD2	I/O port or analog input selection 0: P92 1: AN14
1-0	-	-

■ Port 9 Input Mode Register (P9IMD: 0x03EC9)

MN101EFA7/MN101EFA2

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	P9IMD4	-	-	-	-
At reset	-	-	-	0	-	-	-	-
Access	-	-	-	R/W	-	-	-	-

bp	Flag	Description
7-5	-	-
4	P9IMD4	I/O port or analog input selection 0: P94 1: AN8
3-0	-	-

- ■ Port 9 Pull-up/pull-down Resistor Selection Register 2(SELUD2: 0x03EBF)

MN101EFA8/MN101EFA3

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	SELUDB	-	SELUD9	-
At reset	-	-	-	-	0	-	0	-
Access	-	-	-	-	R/W	-	R/W	-

bp	Flag	Description
7-4	-	-
3	SELUDB	Pull-up/pull-down selection (Port B) 0: pull-up 1: pull-down
2	-	-
1	SELUD9	Pull-up/pull-down selection (Port 9) 0: pull-up 1: pull-down
0	-	-

- ■ Port 9 Pull-up/pull-down Resistor Selection Register 2 (SELUD2: 0x03EBF)

MN101EFA7/MN101EFA2

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	-	-	SELUD9	-
At reset	-	-	-	-	-	-	0	-
Access	-	-	-	-	-	-	R/W	-

bp	Flag	Description
7-2	-	-
1	SELUD9	Pull-up/pull-down selection (Port 9) 0: pull-up 1: pull-down
0	-	-

5.11.3 Block Diagram

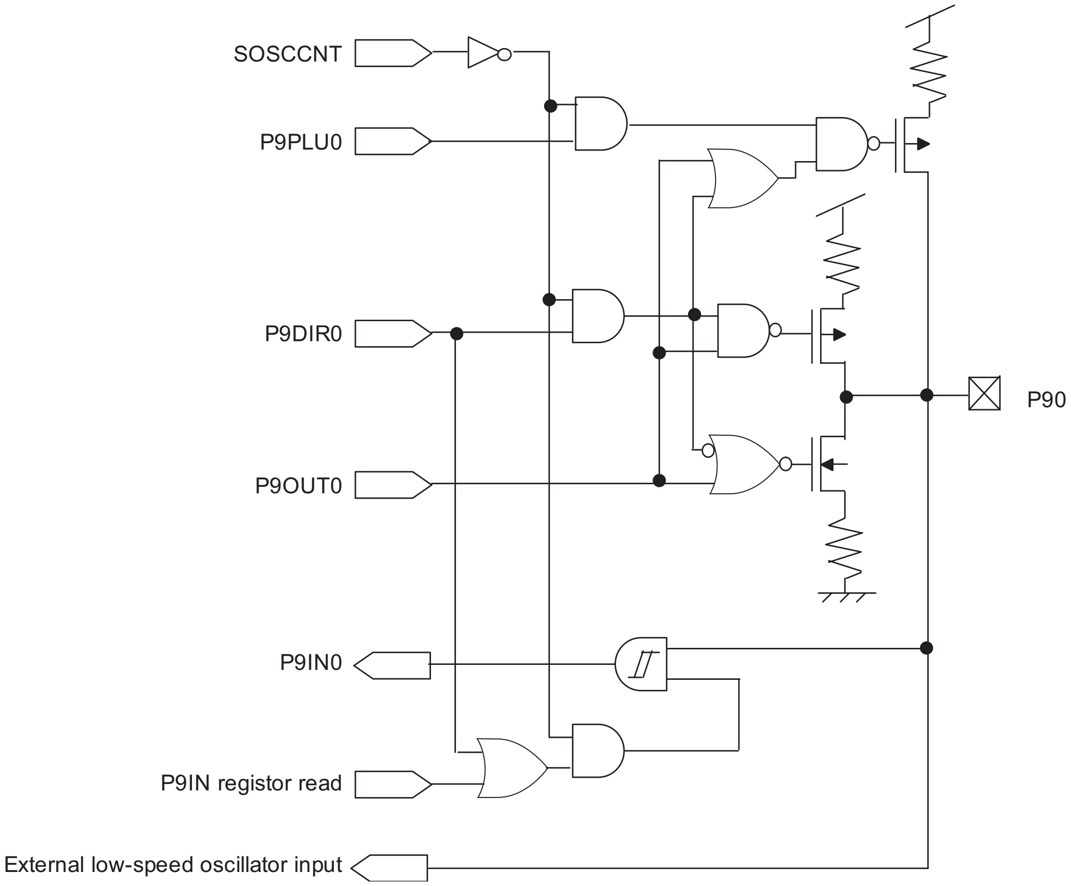


Figure:5.11.1 P90 Block Diagram (MN101EFA8/A7/A3/A2)

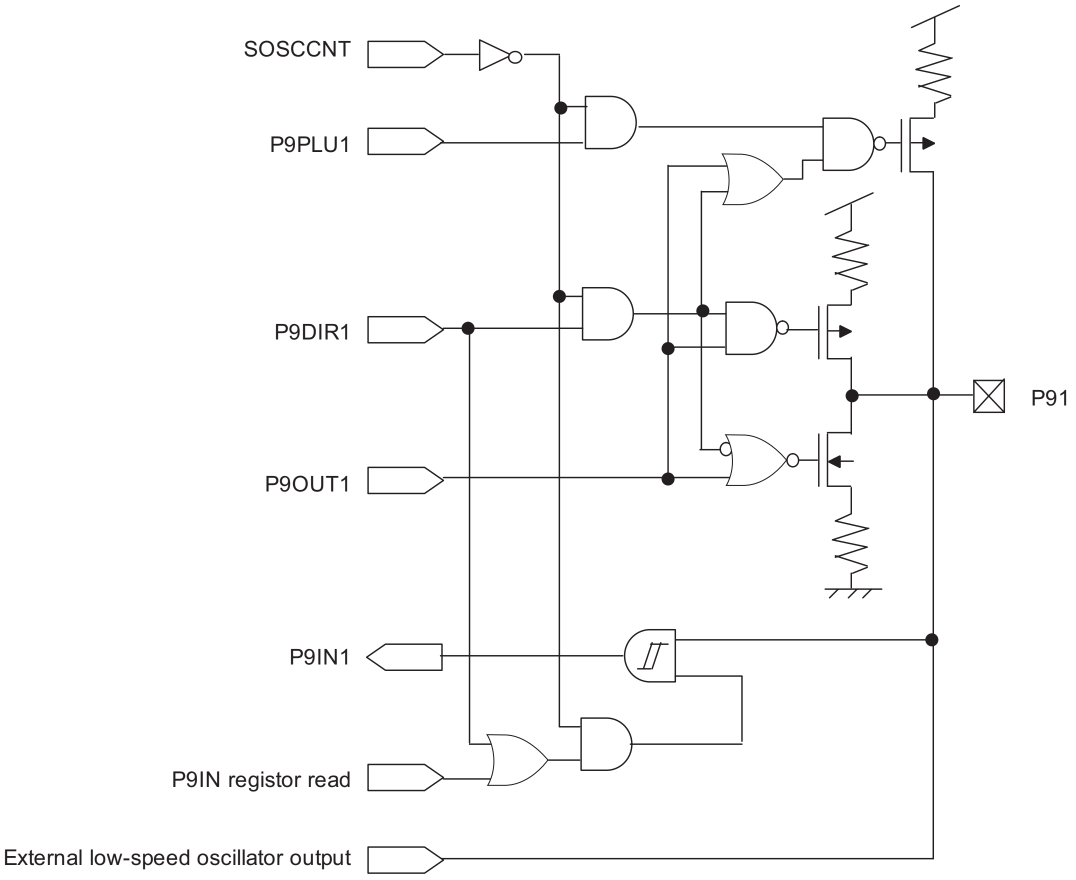


Figure:5.11.2 P91 Block Diagram (MN101EFA8/A7/A3/A2)

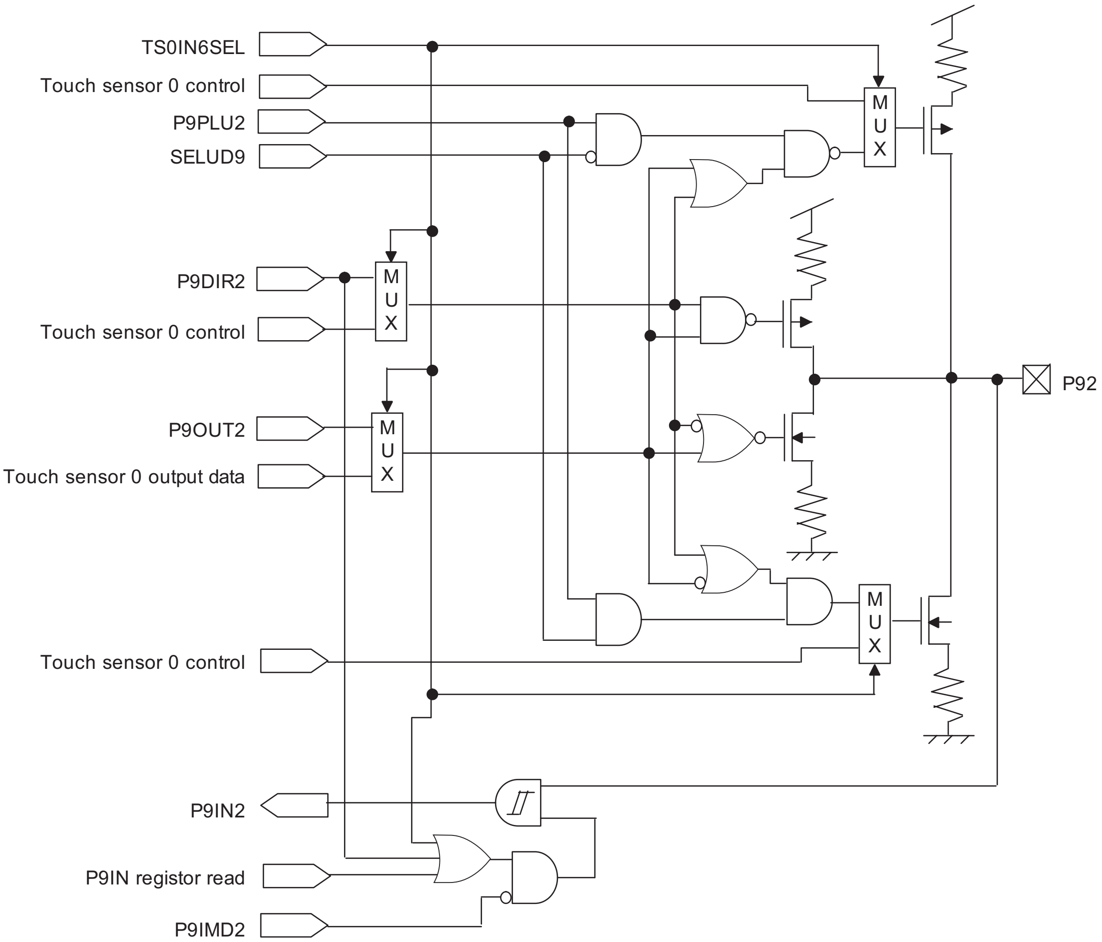


Figure:5.11.3 P92 Block Diagram (MN101EFA8)

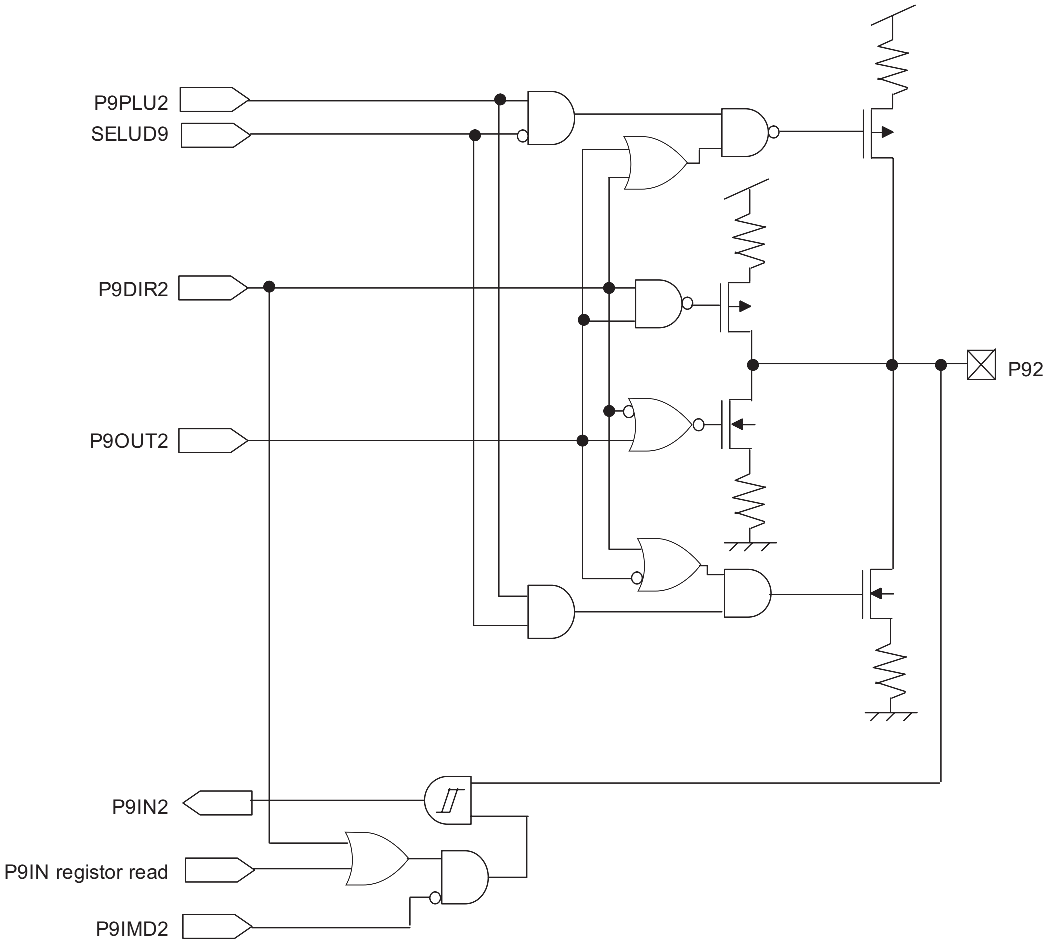


Figure:5.11.4 P92 Block Diagram (MN101EFA3)

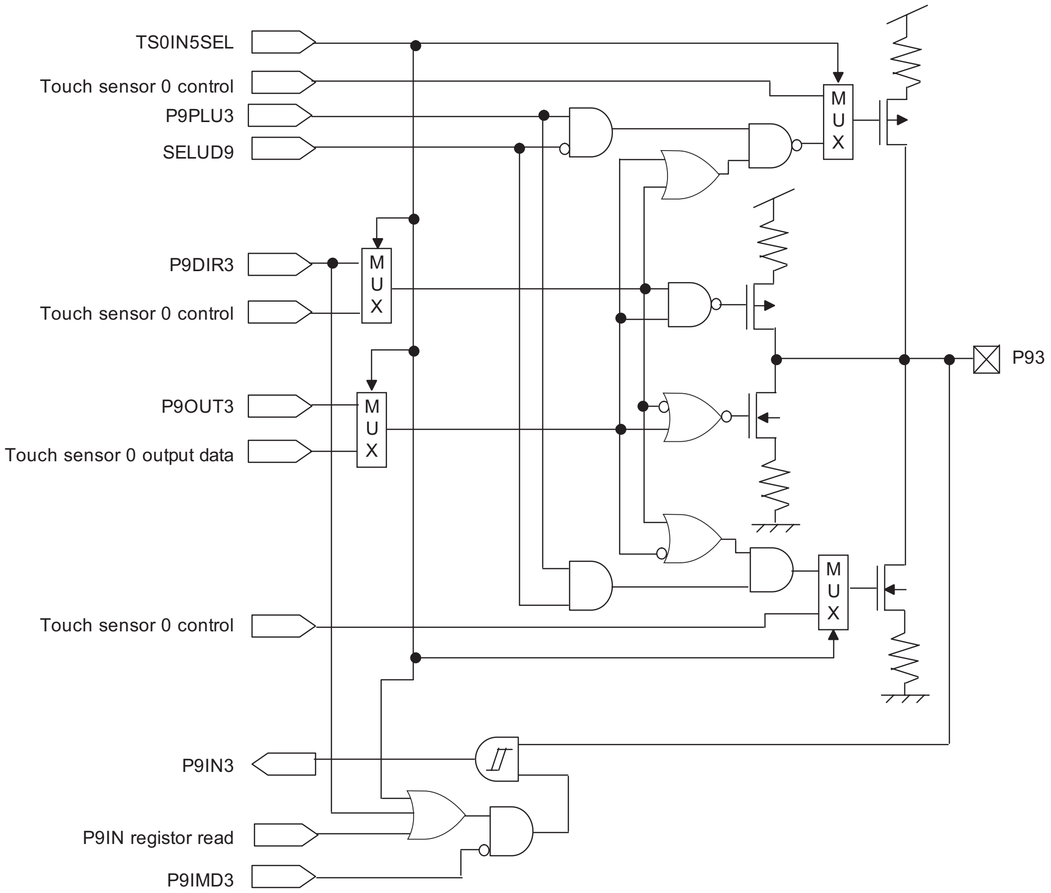


Figure:5.11.5 P93 Block Diagram (MN101EFA8)

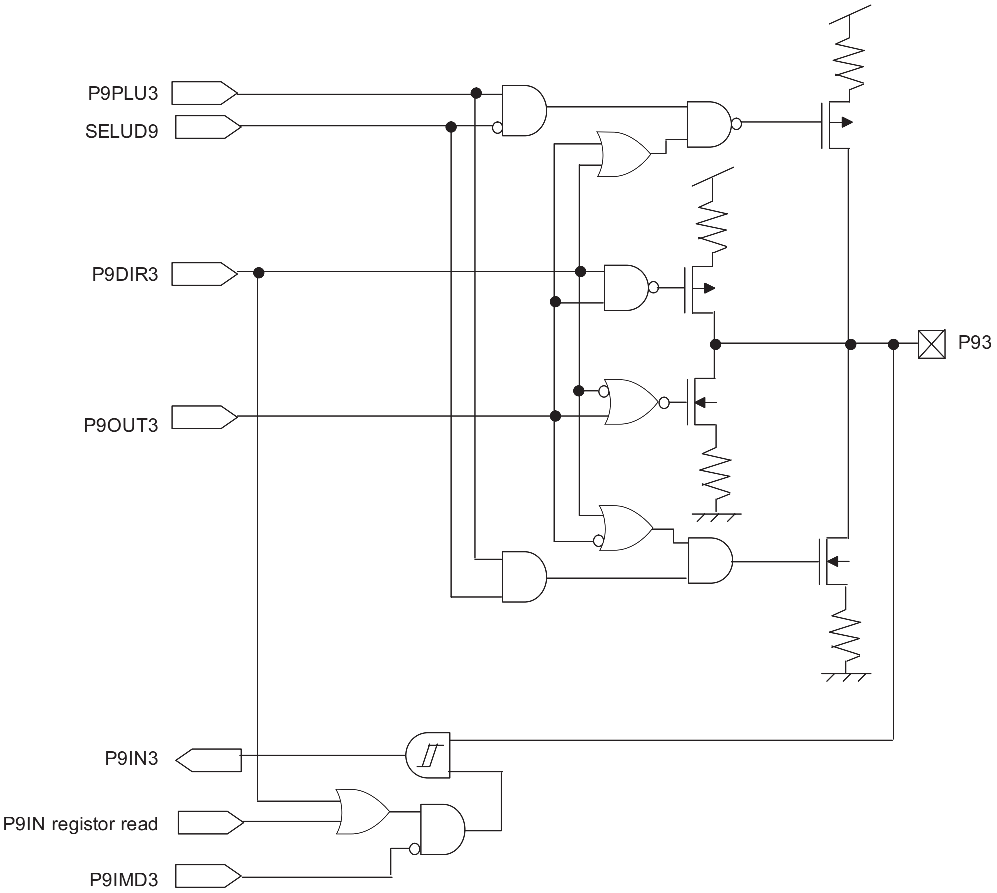
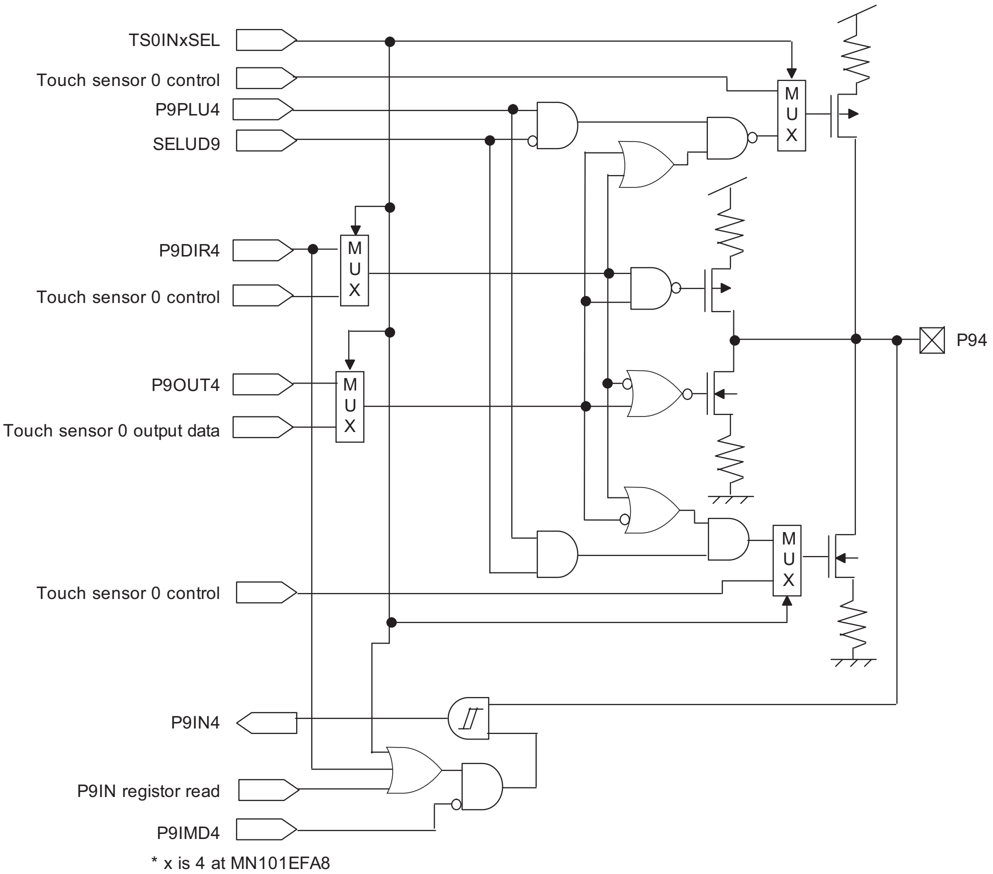


Figure:5.11.6 P93 Block Diagram (MN101EFA3)



* x is 4 at MN101EFA8
x is 0 at MN101EFA7

Figure:5.11.7 P94 Block Diagram (MN101EFA8/A7)

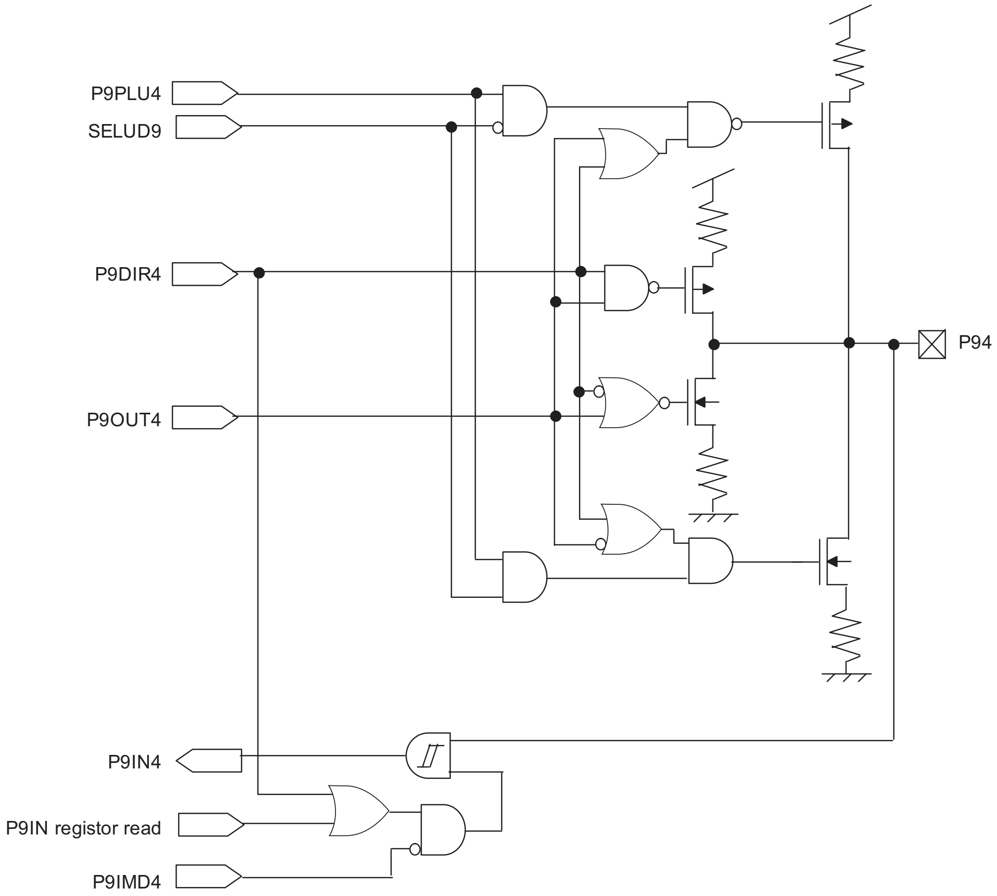


Figure:5.11.8 P94 Block Diagram (MN101EFA3)

5.12 Port A

5.12.1 Description

■ General Pin Setup

To output data to pin, set the control flag of PADIR register to "1" and write data to PAOUT register.

To read input data of pin, set the control flag of PADIR register to "0" and read the value of PAIN register.

Each bit can be set individually to either an input or output by PADIR register. The control flag of PADIR register is set to "1" for output mode, and "0" for input mode.

Each bit can be set individually if pull-up resistor is added or not, by PAPLU register. Set the control flag of PAPLU register to "1" to add pull-up resistor.

Each bit can be selected individually as input mode by PAIMD register. The control flag of PAIMD register is set to "1" to input the special function data and the value read from PAIN register is read to undefined, and "0" to use as the general port.

■ Special Function Pin Setup

PA0 to PA7 are also used as analog input pin. Input mode for each bit can be selected by PAIMD register. When PA0 is used as analog input pin, the value read from the port A input register is undefined.

PA0 is also used as the I/O pin of timer 0.

PA1 is also used as the I/O pin of timer 1.

PA2 is also used as the I/O pin of timer 2.

PA3 is also used as the I/O pin of timer 3.

PA5 is also used as the I/O pin of timer 7.

PA6 is also used as the I/O pin of timer 8.

PA7 is also used as the I/O pin of timer 9.

Each bit for the I/O mode can be selected by PAOMD register. PAOMD register is set to "1" to in/output the special function data, and "0" to use as the general port.

PA0 to PA7 are also used as the LED0 to LED7 output pins.
Output mode of each bit can be set individually by LEDCNT register.
When LEDCNT register is set to "1", PA0 to PA7 are the large current output pins (Nch-Tr. side), and when set to "0", PA0 to PA7 are used as the normal current output pins.
The general port can output a large current in combination with the setting of LEDCNT register.

Table:5.12.1 The Port A Special Function Pins

Table remarks √: With function -: Without function

Pins	Special Functions	MN101EFA8	MN101EFA3	MN101EFA7	MN101EFA2
PA0	AN0	√	√	√	√
	LED0	√	√	√	√
	TM0IOA	√	√	√	√
PA1	AN1	√	√	√	√
	LED1	√	√	√	√
	TM1IOA	√	√	√	√
PA2	AN2	√	√	√	√
	LED2	√	√	√	√
	TM2IOA	√	√	√	√
PA3	AN3	√	√	√	√
	LED3	√	√	√	√
	TM3IOA	√	√	√	√
PA4	AN4	√	√	√	√
	LED4	√	√	√	√
PA5	AN5	√	√	√	√
	LED5	√	√	√	√
	TM7IOA	√	√	√	√
PA6	AN6	√	√	√	√
	LED6	√	√	√	√
	TM8IOA	√	√	√	√
PA7	AN7	√	√	√	√
	LED7	√	√	√	√
	TM9IOA	√	√	√	√

5.12.2 Registers

- Port A output register (PAOUT: 0x03E7A)

bp	7	6	5	4	3	2	1	0
Flag	PAOUT7	PAOUT6	PAOUT5	PAOUT4	PAOUT3	PAOUT2	PAOUT1	PAOUT0
At reset	X	X	X	X	X	X	X	X
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-0	PAOUT7-0	Output data 0: Output "Low" (V_{SS} level) 1: Output "High" (V_{DD5} level)

- Port A Input Register (PAIN: 0x03E8A)

bp	7	6	5	4	3	2	1	0
Flag	PAIN7	PAIN6	PAIN5	PAIN4	PAIN3	PAIN2	PAIN1	PAIN0
At reset	X	X	X	X	X	X	X	X
Access	R	R	R	R	R	R	R	R

bp	Flag	Description
7-0	PAIN7-0	Input data 0: Pin is "Low" (V_{SS} level) 1: Pin is "High" (V_{DD5} level)

- Port A Direction Control Register (PADIR: 0x03E9A)

bp	7	6	5	4	3	2	1	0
Flag	PADIR7	PADIR6	PADIR5	PADIR4	PADIR3	PADIR2	PADIR1	PADIR0
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-0	PADIR7-0	I/O mode selection 0: Input mode 1: Output mode

- Port A Pull-up Resistor Control Register (PAPLU: 0x03EAA)

bp	7	6	5	4	3	2	1	0
Flag	PAPLU7	PAPLU6	PAPLU5	PAPLU4	PAPLU3	PAPLU2	PAPLU1	PAPLU0
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-0	PAPLU7-0	Pull-up resistor selection 0: Not added 1: Added

- Port A Output Mode Register (PAOMD: 0x03EBA)

bp	7	6	5	4	3	2	1	0
Flag	PAOMD7	PAOMD6	PAOMD5	Reserved	PAOMD3	PAOMD2	PAOMD1	PAOMD0
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7	PAOMD7	I/O port or special function selection 0: PA7 1: TM9IOA
6	PAOMD6	I/O port or special function selection 0: PA6 1: TM8IOA
5	PAOMD5	I/O port or special function selection 0: PA5 1: TM7IOA
4	Reserved	Always set to “0“.
3	PAOMD3	I/O port or special function selection 0: PA3 1: TM3IOA
2	PAOMD2	I/O port or special function selection 0: PA2 1: TM2IOA
1	PAOMD1	I/O port or special function selection 0: PA1 1: TM1IOA
0	PAOMD0	I/O port or special function selection 0: PA0 1: TM0IOA

- Port A Input Mode Register (PAIMD: 0x03ECA)

bp	7	6	5	4	3	2	1	0
Flag	PAIMD7	PAIMD6	PAIMD5	PAIMD4	PAIMD3	PAIMD2	PAIMD1	PAIMD0
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7	PAIMD7	I/O port or analog input selection 0: PA7 1: AN7
6	PAIMD6	I/O port or analog input selection 0: PA6 1: AN6
5	PAIMD5	I/O port or analog input selection 0: PA5 1: AN5
4	PAIMD4	I/O port or analog input selection 0: PA4 1: AN4
3	PAIMD3	I/O port or analog input selection 0: PA3 1: AN3
2	PAIMD2	I/O port or analog input selection 0: PA2 1: AN2
1	PAIMD1	I/O port or analog input selection 0: PA1 1: AN1
0	PAIMD0	I/O port or analog input selection 0: PA0 1: AN0

■ Port LED Control Register (LEDCNT: 0x03EE0)

bp	7	6	5	4	3	2	1	0
Flag	LEDCNT7	LEDCNT6	LEDCNT5	LEDCNT4	LEDCNT3	LEDCNT2	LEDCNT1	LEDCNT0
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7	LEDCNT7	LED7 (large current output) selection 0: Normal output 1: LED7 (large current output)
6	LEDCNT6	LED6 (large current output) selection 0: Normal output 1: LED6 (large current output)
5	LEDCNT5	LED5 (large current output) selection 0: Normal output 1: LED5 (large current output)
4	LEDCNT4	LED4 (large current output) selection 0: Normal output 1: LED4 (large current output)
3	LEDCNT3	LED3 (large current output) selection 0: Normal output 1: LED3 (large current output)
2	LEDCNT2	LED2 (large current output) selection 0: Normal output 1: LED2 (large current output)
1	LEDCNT1	LED1 (large current output) selection 0: Normal output 1: LED1 (large current output)
0	LEDCNT0	LED0 (large current output) selection 0: Normal output 1: LED0 (large current output)

5.12.3 Block Diagram

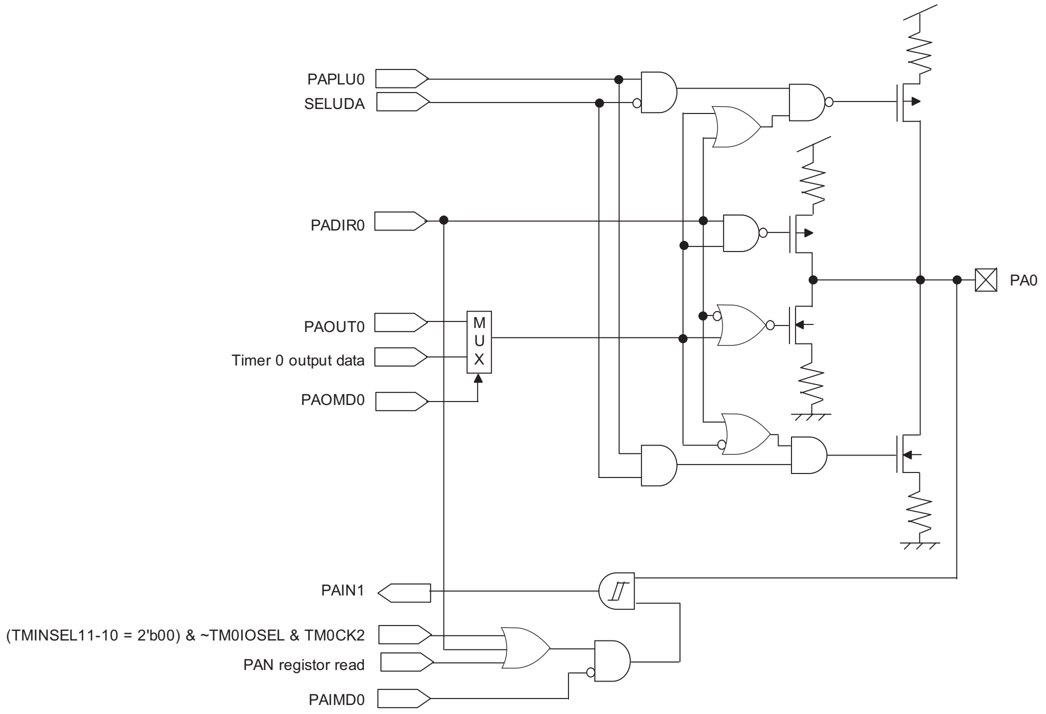


Figure:5.12.1 PA0 Block Diagram (MN101EFA8/A7/A3/A2)

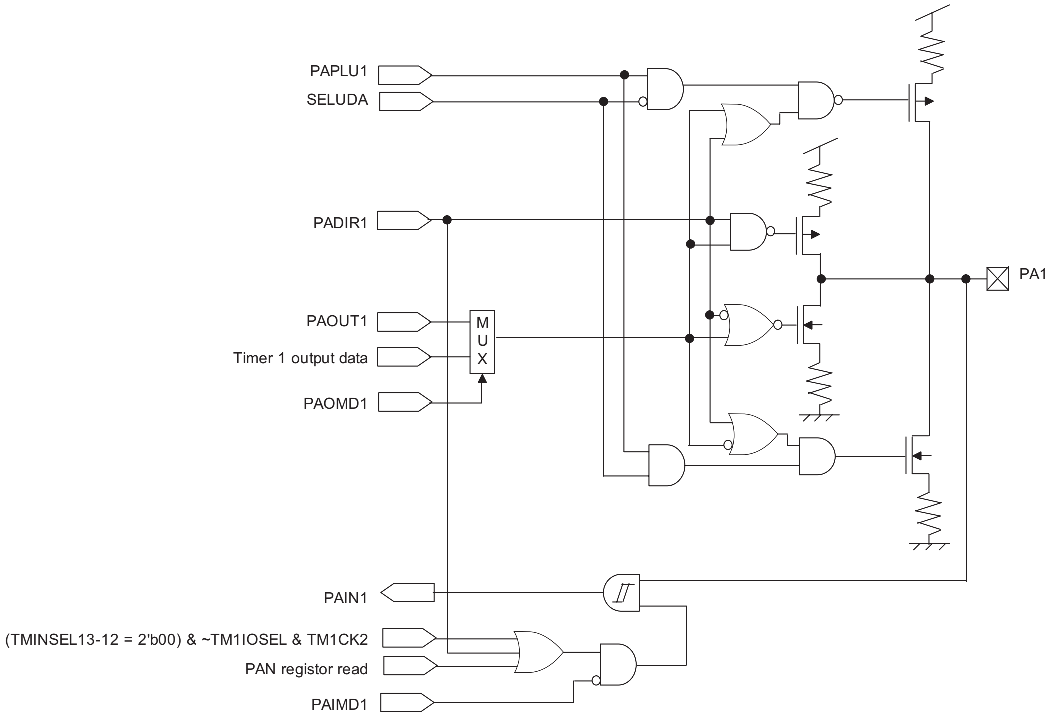


Figure:5.12.2 PA1 Block Diagram (MN101EFA8/A7/A3/A2)

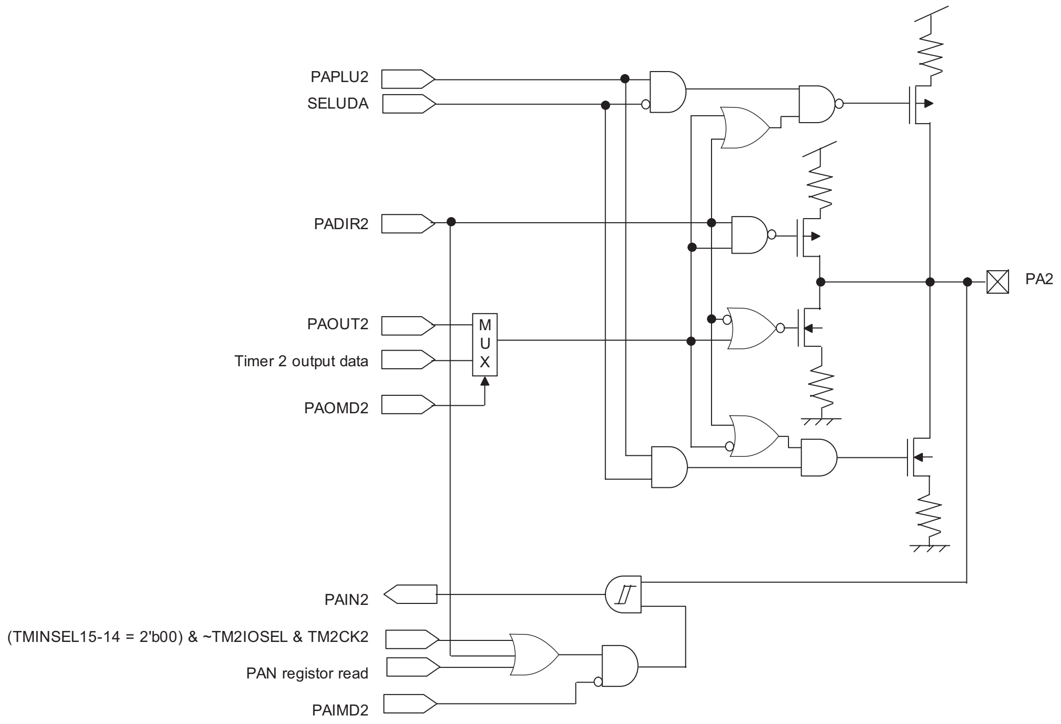


Figure:5.12.3 PA2 Block Diagram (MN101EFA8/A7/A3/A2)

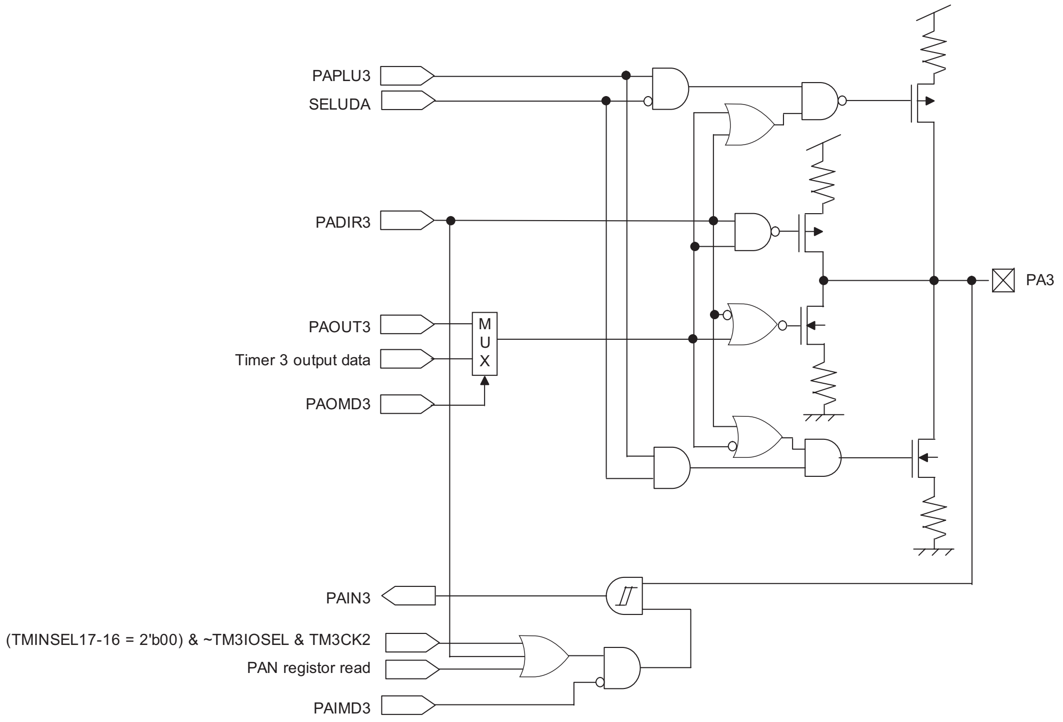


Figure:5.12.4 PA3 Block Diagram (MN101EFA8/A7/A3/A2)

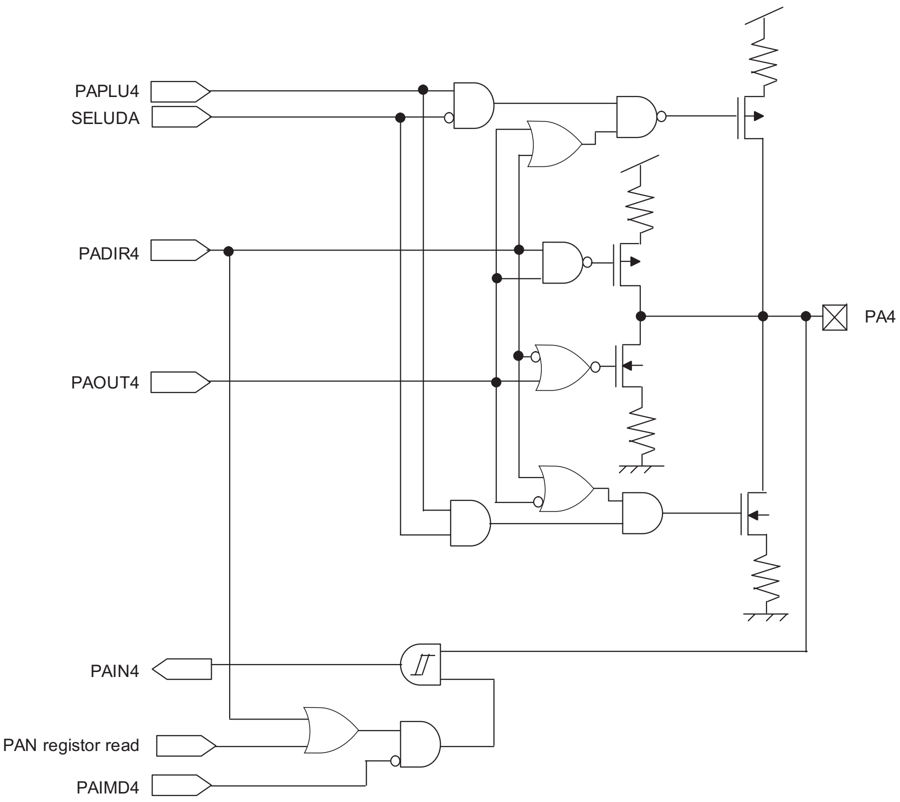


Figure:5.12.5 PA4 Block Diagram (MN101EFA8/A7/A3/A2)

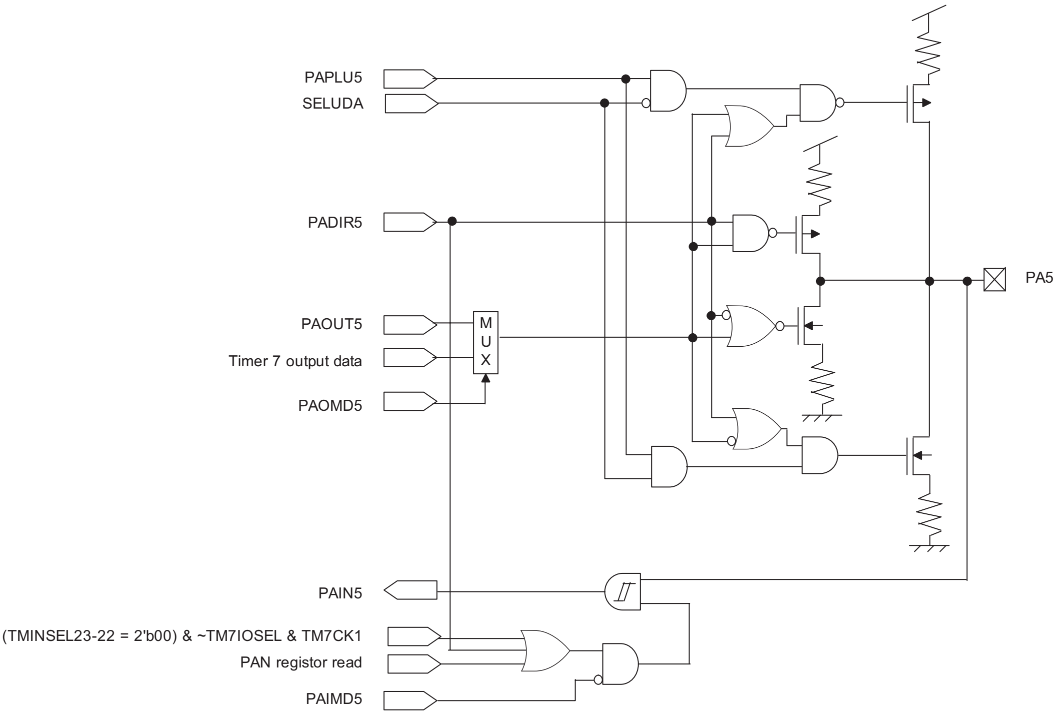


Figure:5.12.6 PA5 Block Diagram (MN101EFA8/A7/A3/A2)

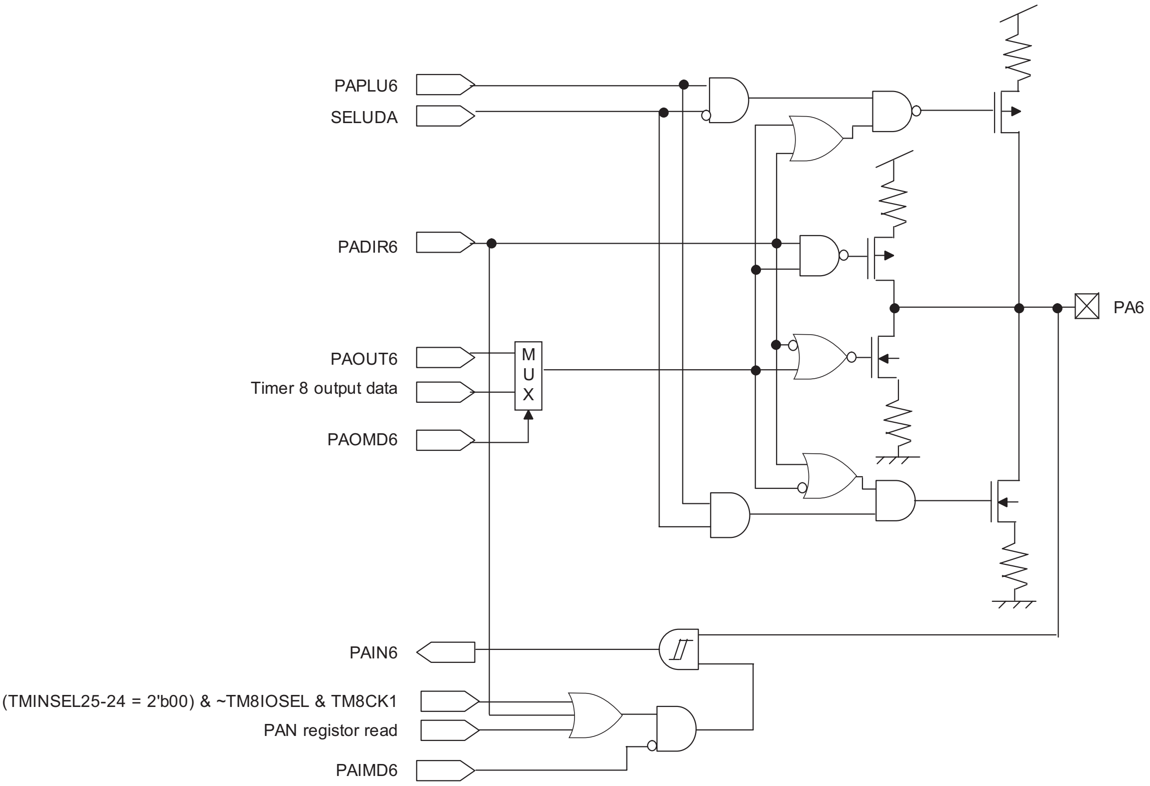


Figure:5.12.7 PA6 Block Diagram (MN101EFA8/A7/A3/A2)

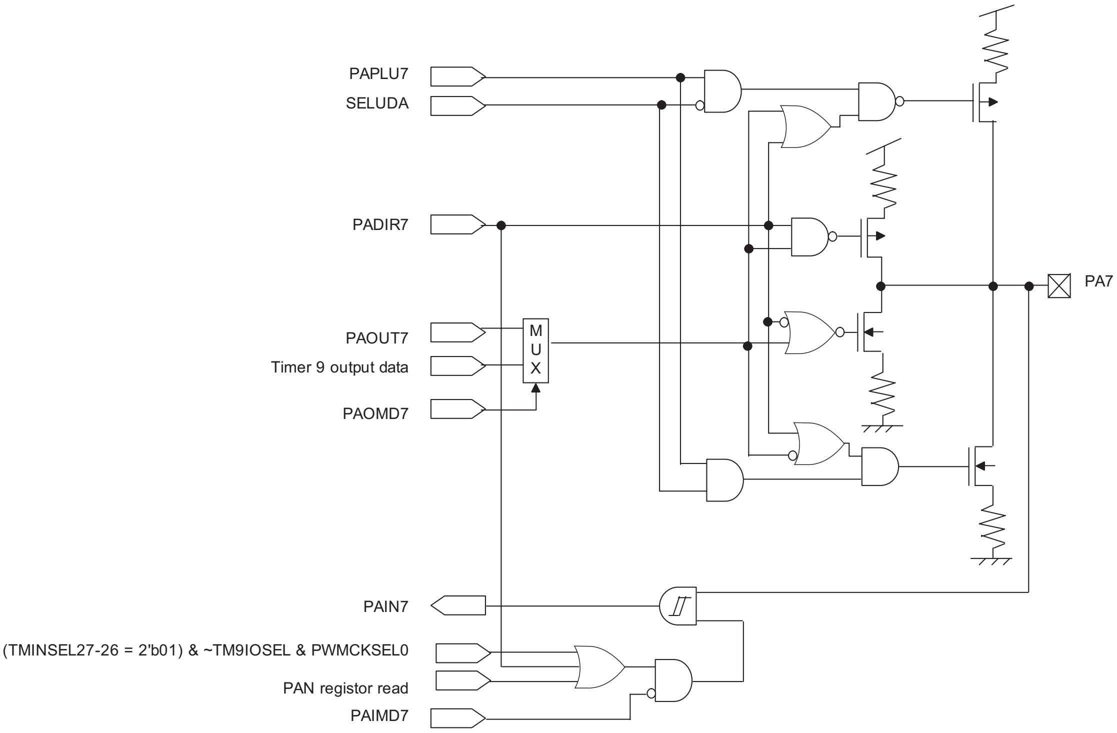


Figure:5.12.8 PA7 Block Diagram (MN101EFA8/A7/A3/A2)

5.13 Port B

5.13.1 Description

MN101EFA7/A2 do not include the Port B.

■ General Pin Setup

To output data to pin, set the control flag of PBDIR register to "1" and write data to PBOUT register.

To read input data of pin, set the control flag of PBDIR register to "0" and read the value of PBIN register.

Each bit can be set individually to either an input or output by PBDIR register. The control flag of PBDIR register is set to "1" for output mode, and "0" for input mode.

Each bit can be set individually if pull-up resistor is added or not, by PBPLUD register. Set the control flag of PBPLUD register to "1" to add pull-up resistor.
Port B can be selected to add pull-up or pull-down resistor by the SELUDB flag of SELUD2 register.

Each bit can be selected individually as input mode by PBIMD register. PBIMD register is set to "1" to input the special function data and the value read from PBIN register is read to undefined, and "0" to use as the general port.

■ Special Function Pin Setup

PB0 to PB3 are also used as analog input pin. Input mode for each bit can be selected by PBIMD register. When these pins are used as analog input pin, the value read from the port B input register is read to be "0".

(This function is equipped in MN101EFA8.)
PB0 to PB3 are also used as input pins for touch sensor timer.
Set "Used" to corresponding channel by TS0TCHSEL register. Refer to [Chapter XV Touch Sensor Timer].

Table:5.13.1 The Port B Special Function Pins

Table remarks √: With function -: Without function

Pins	Special Functions	MN101EFA8	MN101EFA3	MN101EFA7	MN101EFA2
PB0	AN8	√	√	-	-
	TSIN00	√	-	-	-
PB1	AN9	√	√	-	-
	TSIN01	√	-	-	-
PB2	AN10	√	√	-	-
	TSIN02	√	-	-	-
PB3	AN11	√	√	-	-
	TSIN03	√	-	-	-

5.13.2 Registers

- Port B Output Register (PBOUT: 0x03E7B)

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	PBOUT3	PBOUT2	PBOUT1	PBOUT0
At reset	-	-	-	-	X	X	X	X
Access	-	-	-	-	R/W	R/W	R/W	R/W

bp	Flag	Description
7-4	-	-
3-0	PBOUT3-0	Output data 0: Output "Low" (V_{SS} level) 1: Output "High" (V_{DD5} level)

- Port B Input Register (PBIN: 0x03E8B)

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	PBIN3	PBIN2	PBIN1	PBIN0
At reset	-	-	-	-	X	X	X	X
Access	-	-	-	-	R	R	R	R

bp	Flag	Description
7-4	-	-
3-0	PBIN3-0	Input data 0: Pin is "Low" (V_{SS} level) 1: Pin is "High" (V_{DD5} level)

- Port B Direction Control Register (PBDIR: 0x03E9B)

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	PBDIR3	PBDIR2	PBDIR1	PBDIR0
At reset	-	-	-	-	0	0	0	0
Access	-	-	-	-	R/W	R/W	R/W	R/W

bp	Flag	Description
7-0	-	-
3-0	PBDIR3-0	I/O mode selection 0: Input mode 1: Output mode

- Port B Pull-up/pull-down Resistor Selection Register (PBPLUD: 0x03EAB)

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	PBPLUD3	PBPLUD2	PBPLUD1	PBPLUD0
At reset	-	-	-	-	0	0	0	0
Access	-	-	-	-	R/W	R/W	R/W	R/W

bp	Flag	Description
7-4	-	-
3-0	PBPLUD3-0	Pull-up/pull-down resistor selection) 0: No added 1: Added

- Port B Input Mode Register (PBIMD: 0x03ECB)

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	PBIMD3	PBIMD2	PBIMD1	PBIMD0
At reset	-	-	-	-	0	0	0	0
Access	-	-	-	-	R/W	R/W	R/W	R/W

bp	Flag	Description
7-4	-	-
3	PAIMD3	I/O port or special function selection 0: PB3 1: AN11
2	PAIMD2	I/O port or special function selection 0: PB2 1: AN10
1	PAIMD1	I/O port or special function selection 0: PB1 1: AN9
0	PAIMD0	I/O port or special function selection 0: PB0 1: AN8

5.13.3 Block Diagram

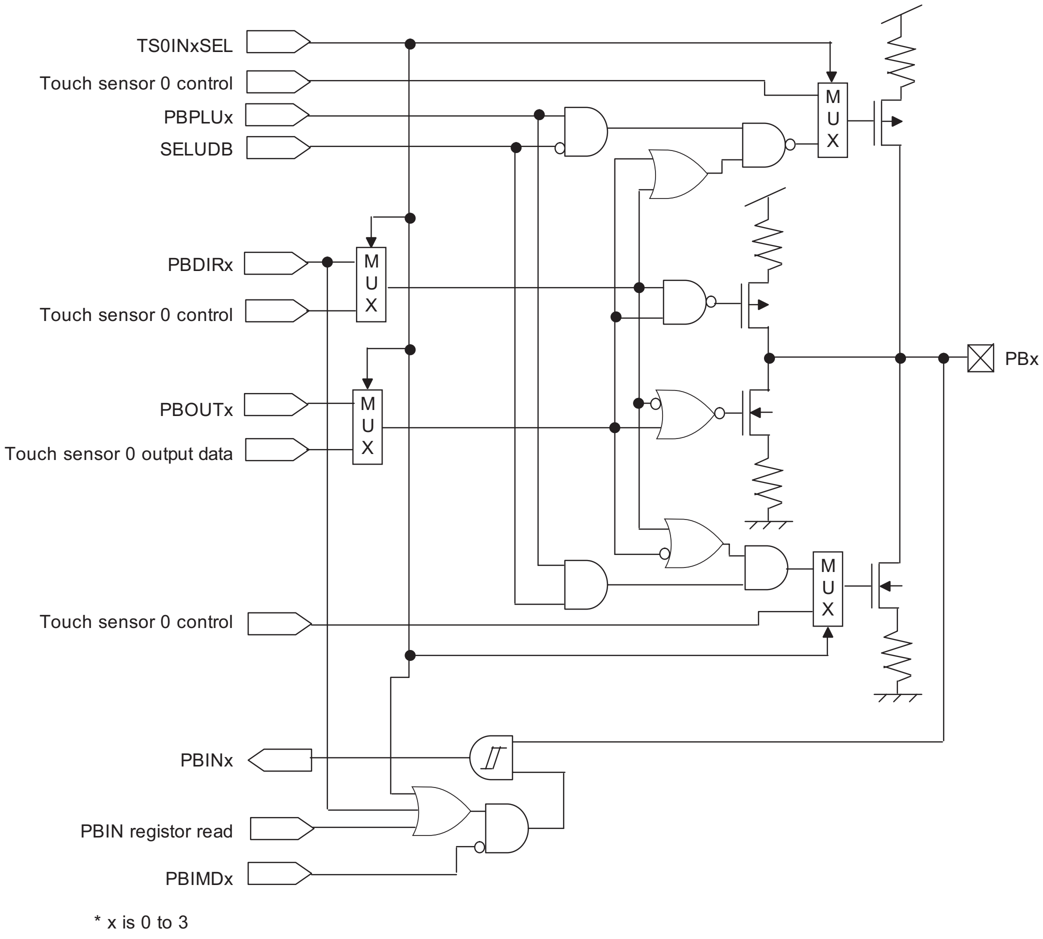
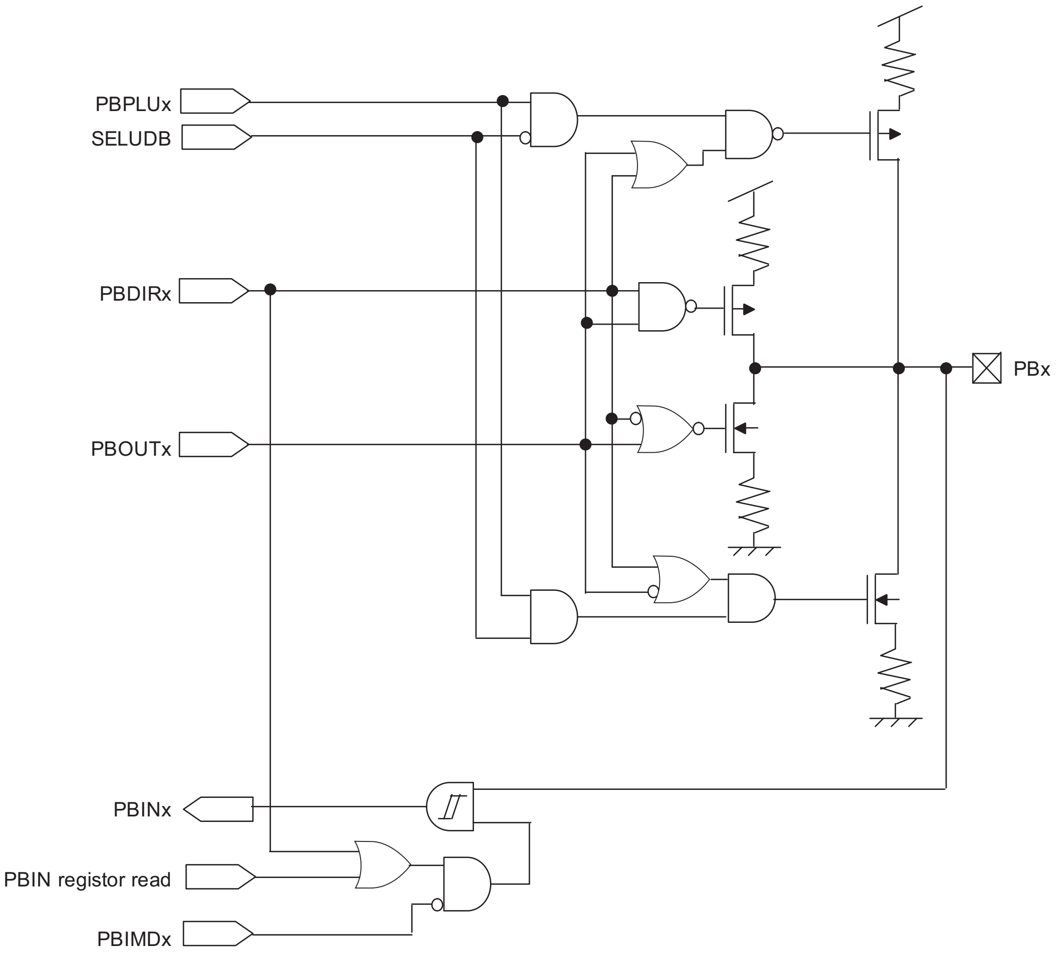


Figure:5.13.1 PBx Block Diagram (MN101EFA8)



* x is 0 to 3

Figure:5.13.2 PBx Block Diagram (MN101EFA3)

Chapter 6 8-bit Timers

6.1 Overview

This LSI contains five 8-bit timers/baud rate timers (Timer 0, 1, 2 and 3). Timer 0 and 1 or Timer 2 and 3 can be cascaded to form 16-bit timer. Under cascade connection, Timer 0, 2 forms the lower 8 bits of the 16-bit timer while Timer 1 forms the upper 8 bits of the 16-bit timer. In addition, cascading Timer 0 to 2 form a 24-bit counter, and cascading Timer 0 to 3 form a 32-bit counter.

8-bit timer consists of two prescalers which can be used simultaneously. Each prescaler counts fpll_div and fs as the base clock. Configurations of hard ware are shown below.

Prescaler 0 (based on fpll-div)	7-bit prescaler
Prescaler 1 (based on fs)	3-bit prescaler

Prescaler 0 outputs fpll_div/4, fpll_div/16, fpll_div/32, fpll_div/64 and fpll_div/128.
Prescaler 1 outputs fs/2, fs/4 and fs/8.

Divided clock of fpll_div or fs can be selected as the clock source for each timer by using the prescaler output. Pins to be used can be switched to TMnIOA/TMnIOB.

Table:6.1.1 8-bit Timer Pin Functions

Table remarks √: With function -: Without function

Functions	Pin Name	MN101EFA8/A3	MN101EFA7/A2
TM0IOA	PA0	√	√
TM0IOB	P04	√	√
TM1IOA	PA1	√	√
TM1IOB	P62	√	√
TM2IOA	PA2	√	-
TM2IOB	P04	√	√
TM3IOA	PA3	√	√
TM3IOB	P63	√	√

In this manual, if there is not much difference in the function between Pin A and B, “A” and “B” of the pin names are omitted.

When changing fpll-div frequency by bp7 to bp4 of OSCCNT register, it should be executed after the 8-bit timer is stopped.

6.1.1 Functions

Table:6.1.2 shows functions that can be used with each timer.

Table:6.1.2 Timer Functions

<table>
<tr><th></th><th>Timer 0 (8bit)</th><th>Timer 1 (8bit)</th><th>Timer 2 (8bit)</th><th>Timer 3 (8bit)</th></tr>
<tr><td>Interrupt source</td><td>TM0IRQ</td><td>TM1IRQ</td><td>TM2IRQ</td><td>TM3IRQ</td></tr>
<tr><td>Timer operation</td><td>√</td><td>√</td><td>√</td><td>√</td></tr>
<tr><td rowspan="2">Event count</td><td>TM0IOA input</td><td>TM1IOA input</td><td>TM2IOA input</td><td>TM3IOA input</td></tr>
<tr><td>TM0IOB input</td><td>TM1IOB input</td><td>TM2IOB input</td><td>TM3IOB input</td></tr>
<tr><td rowspan="2">Timer pulse output</td><td>TM0IOA output</td><td>TM1IOA output</td><td>TM2IOA output</td><td>TM3IOA output</td></tr>
<tr><td>TM0IOB output</td><td>TM1IOB output</td><td>TM2IOB output</td><td>TM3IOB output</td></tr>
<tr><td rowspan="2">PWM output</td><td>TM0IOA output</td><td rowspan="2">-</td><td>TM2IOA output</td><td rowspan="2">-</td></tr>
<tr><td>TM0IOB output</td><td>TM2IOB output</td></tr>
<tr><td>PWM output with additional pulses</td><td>√</td><td>-</td><td>√</td><td>-</td></tr>
<tr><td>Serial transfer clock output</td><td>Serial 0, 1, 2, 4</td><td>Serial 0, 1, 2, 4</td><td>Serial 0, 1, 2, 4</td><td>Serial 0, 1, 2, 4</td></tr>
<tr><td>Pulse width measurement</td><td>External interrupt 0 (P20/IRQ0)</td><td>-</td><td>External interrupt 2 (P22/IRQ2)</td><td>-</td></tr>
<tr><td rowspan="3">Cascade Connection</td><td colspan="2">√</td><td colspan="2">√</td></tr>
<tr><td colspan="3">√</td><td>-</td></tr>
<tr><td colspan="4">√</td></tr>
<tr><td rowspan="14">Clock source</td><td>fpll-div</td><td>fpll-div</td><td>fpll-div</td><td>fpll-div</td></tr>
<tr><td>fpll-div/4</td><td>fpll-div/4</td><td>fpll-div/4</td><td>fpll-div/4</td></tr>
<tr><td>fpll-div/16</td><td>fpll-div/16</td><td>fpll-div/16</td><td>fpll-div/16</td></tr>
<tr><td>fpll-div/32</td><td>fpll-div/32</td><td>fpll-div/32</td><td>fpll-div/32</td></tr>
<tr><td>fpll-div/64</td><td>fpll-div/64</td><td>fpll-div/64</td><td>fpll-div/64</td></tr>
<tr><td>fpll-div/128</td><td>fpll-div/128</td><td>fpll-div/128</td><td>fpll-div/128</td></tr>
<tr><td>fs/2</td><td>fs/2</td><td>fs/2</td><td>fs/2</td></tr>
<tr><td>fs/4</td><td>fs/4</td><td>fs/4</td><td>fs/4</td></tr>
<tr><td>fs/8</td><td>fs/8</td><td>fs/8</td><td>fs/8</td></tr>
<tr><td>fx</td><td>fx</td><td>fx</td><td>fx</td></tr>
<tr><td>TM0IO input</td><td>TM1IO input</td><td>TM2IO input</td><td>TM3IO input</td></tr>
<tr><td>Synchronous fx</td><td>Synchronous fx</td><td>Synchronous fx</td><td>Synchronous fx</td></tr>
<tr><td>Synchronous TM0IO input</td><td>Synchronous TM1IO input</td><td>Synchronous TM2IO input</td><td>Synchronous TM3IO input</td></tr>
<tr><td>Timer A output</td><td>Timer A output</td><td>Timer A output</td><td>Timer A output</td></tr>
</table>

fpll-div: Machine clock (High speed oscillation for peripheral functions)
fx: Machine clock (Low speed oscillation)
fs: System clock

6.1.2 Block Diagram

■ Prescaler Block Diagram

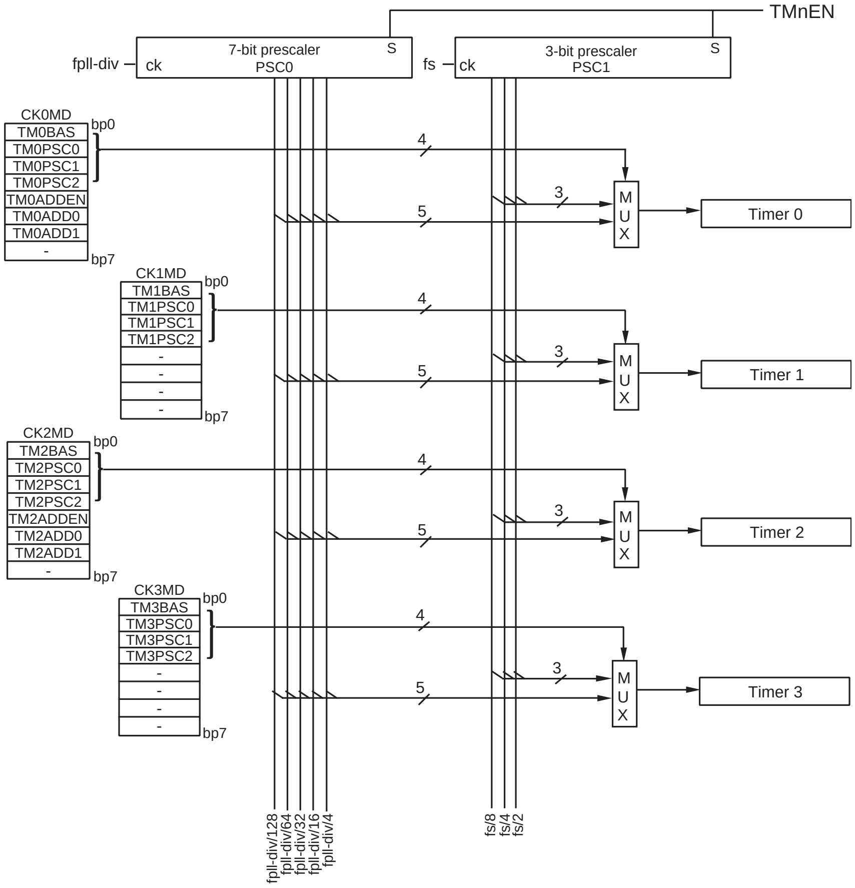


Figure:6.1.1 Prescaler Block Diagram

■ Timer 0 and Timer 1 Block Diagram

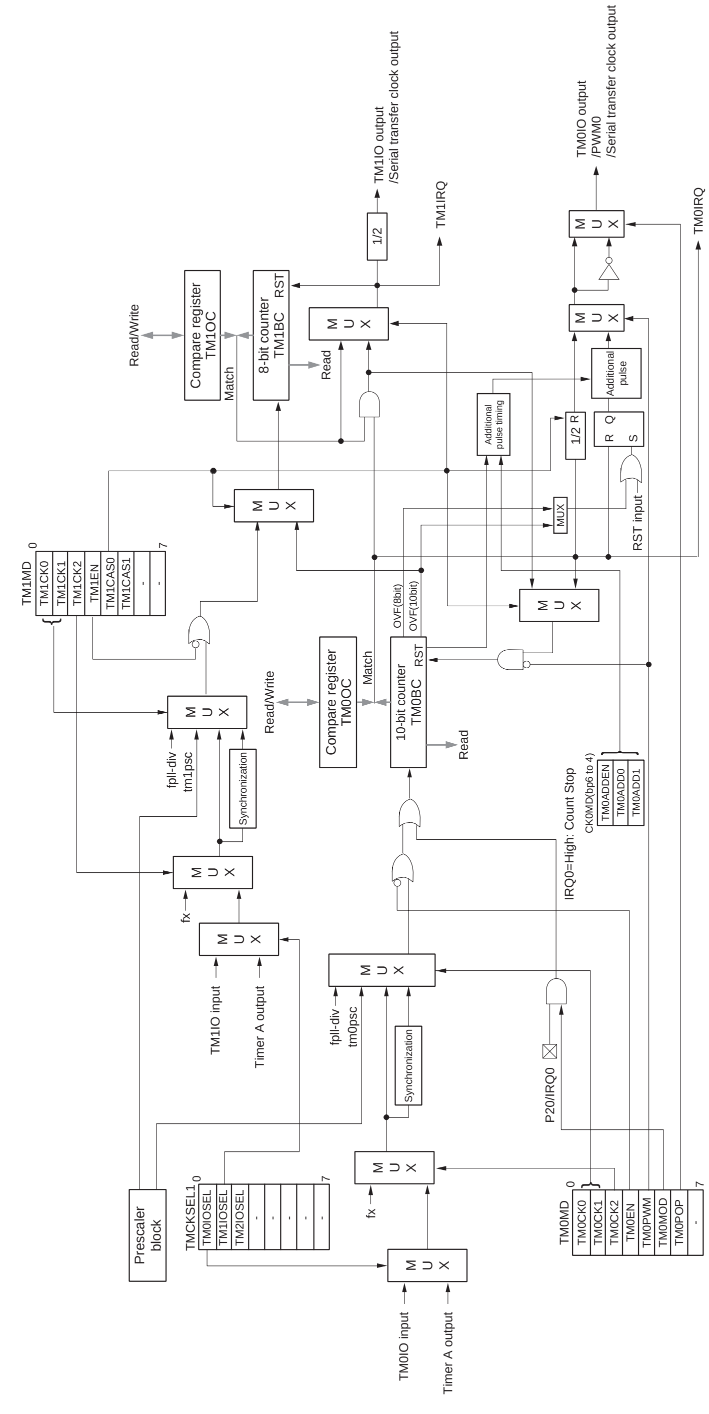


Figure:6.1.2 Timer 0 and Timer 1 Block Diagram

■ Timer 2 and Timer 3 Block Diagram

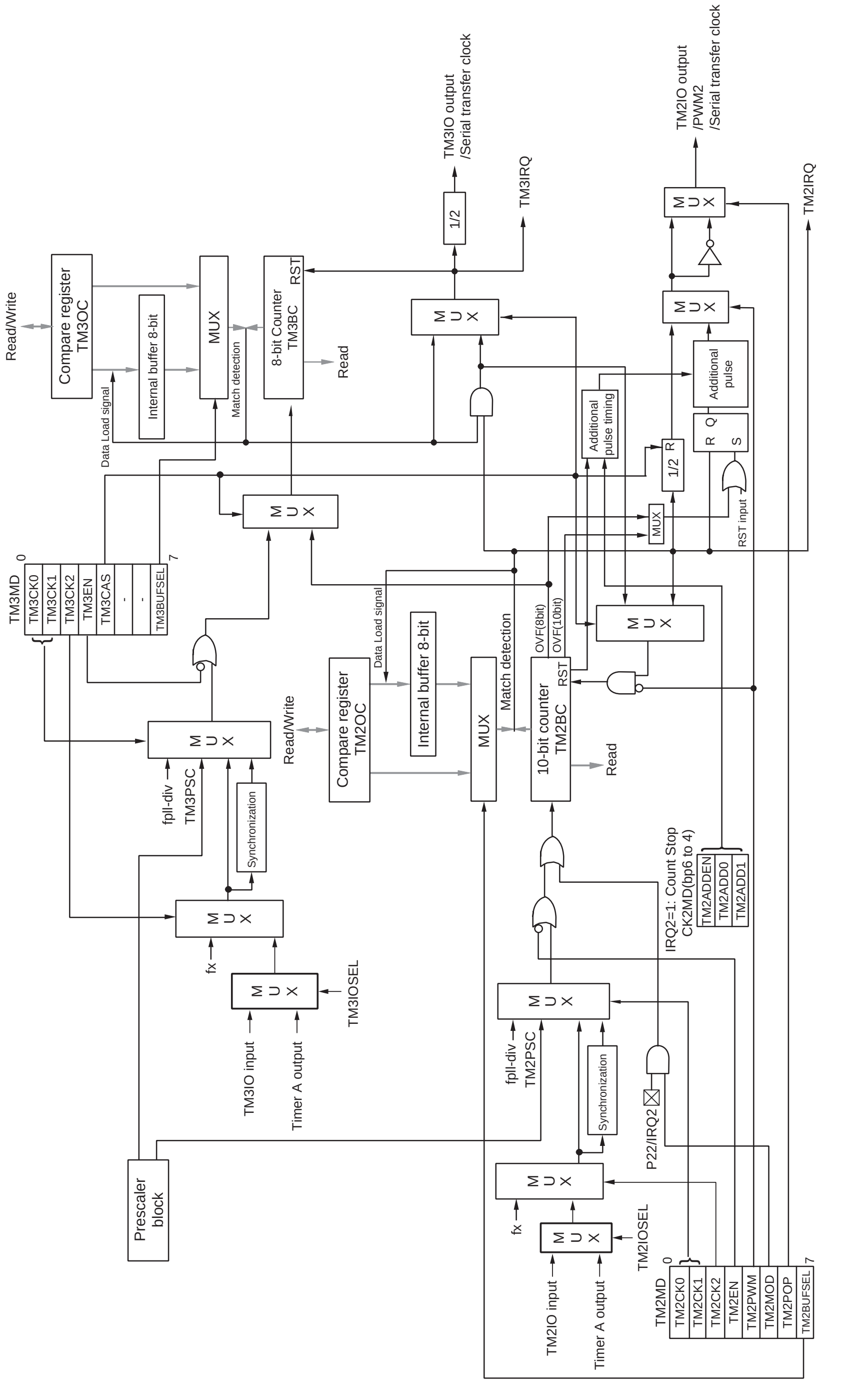


Figure:6.1.3 Timer 2 and Timer 3 Block Diagram

6.2 Control Registers

Timer 0 to 3 consist of binary counter (TMnBC) and compare register (TMnOC).
And they are controlled by mode register (TMnMD).

When the prescaler output is selected as the count clock source of Timer 0 to 3, they should be controlled by prescaler selection register (CKnMD).

6.2.1 Registers

Table:6.2.1 shows registers that control Timer 0 to 3.

Table:6.2.1 8-bit Timer Control Registers

Table remarks √: With function -: Without function

	Register	Address	R/W	Function	Page	MN101 EFA8	MN101 EFA3	MN101 EFA7	MN101 EFA2
Timer 0	TM0BC	0x03F60	R	Timer 0 binary counter	VI-11	√	√	√	√
	TM0OC	0x03F62	R/W	Timer 0 compare register	VI-11	√	√	√	√
	TM0MD	0x03F64	R/W	Timer 0 mode register	VI-12	√	√	√	√
	CK0MD	0x03F66	R/W	Timer 0 prescaler selection register	VI-9	√	√	√	√
	TM0ICR	0x03FEF	R/W	Timer 0 interrupt control register	IV-22	√	√	√	√
	TMCKSEL1	0x03FB0	R/W	Timer clock selection register 1	VI-16	√	√	√	√
	TMINSEL1	0x03FB2	R/W	Timer input selection register 1	VI-17	√	√	√	√
Timer 1	TM1BC	0x03F61	R	Timer 1 binary counter	VI-11	√	√	√	√
	TM1OC	0x03F63	R/W	Timer 1 compare register	VI-11	√	√	√	√
	TM1MD	0x03F65	R/W	Timer 1 mode register	VI-13	√	√	√	√
	CK1MD	0x03F67	R/W	Timer 1 prescaler selection register	VI-10	√	√	√	√
	TM1ICR	0x03FF0	R/W	Timer 1 interrupt control register	IV-22	√	√	√	√
	TMCKSEL1	0x03FB0	R/W	Timer clock selection register 1	VI-16	√	√	√	√
	TMINSEL1	0x03FB2	R/W	Timer input selection register 1	VI-17	√	√	√	√
Timer 2	TM2BC	0x03F68	R	Timer 2 binary counter	VI-11	√	√	√	√
	TM2OC	0x03F6A	R/W	Timer 2 compare register	VI-11	√	√	√	√
	TM2MD	0x03F6C	R/W	Timer 2 mode register	VI-14	√	√	√	√
	CK2MD	0x03F6E	R/W	Timer 2 prescaler selection register	VI-9	√	√	√	√
	TM2ICR	0x03FF1	R/W	Timer 2 interrupt control register	IV-22	√	√	√	√
	TMCKSEL1	0x03FB0	R/W	Timer clock selection register 1	VI-16	√	√	√	√
	TMINSEL1	0x03FB2	R/W	Timer input selection register 1	VI-17	√	√	√	√
Timer 3	TM3BC	0x03F69	R	Timer 3 binary counter	VI-11	√	√	√	√
	TM3OC	0x03F6B	R/W	Timer 3 compare register	VI-11	√	√	√	√
	TM3MD	0x03F6D	R/W	Timer 3 mode register	VI-15	√	√	√	√
	CK3MD	0x03F6F	R/W	Timer 3 prescaler selection register	VI-10	√	√	√	√
	TM3ICR	0x03FF2	R/W	Timer 3 interrupt control register	IV-22	√	√	√	√
	TMCKSEL1	0x03FB0	R/W	Timer clock selection register 1	VI-16	√	√	√	√
	TMINSEL1	0x03FB2	R/W	Timer input selection register 1	VI-17	√	√	√	√

R/W: Readable/Writable
R: Read only

6.2.2 Timer Prescaler Registers

Timer prescaler selection register selects the count clock for 8-bit timer.
The register which selects prescaler output is included in timer prescaler selection register (CKnMD).

- Timer 0, 2 Prescaler Selection Register (CK0MD: 0x03F66, CK2MD: 0x03F6E)

bp	7	6	5	4	3	2	1	0
Flag	-	TMnADD1	TMnADD0	TMnADDEN	TMnPSC2	TMnPSC1	TMnPSC0	TMnBAS
At reset	-	0	0	0	0	0	0	0
Access	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7	-	-
6 to 5	TMnADD1 TMnADD0	Positions of additional pulses (within 4 cycles of PWM basic waveform) 00: No pulse 01: At second cycle 10: At first and third cycle 11: At first, second and third cycle
4	TMnADDEN	PWM output control with additional pulses 0: Prohibited (8-bit PWM output) 1: Permitted
3 to 0	TMnPSC2 TMnPSC1 TMnPSC0 TMnBAS	Clock source selection 0000: fpll-div/4 0010: fpll-div/16 0100: fpll-div/32 0110: fpll-div/64 1XX0: fpll-div/128 0X01: fs/2 0X11: fs/4 1XX1: fs/8

- Timer 1, 3 Prescaler Selection Register (CK1MD: 0x03F67, CK3MD: 0x03F6F)

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	TMnPSC2	TMnPSC1	TMnPSC0	TMnBAS
At reset	-	-	-	-	0	0	0	0
Access	-	-	-	-	R/W	R/W	R/W	R/W

bp	Flag	Description
7 to 4	-	-
3 to 0	TMnPSC2 TMnPSC1 TMnPSC0 TMnBAS	Clock source selection 0000: fpll-div/4 0010: fpll-div/16 0100: fpll-div/64 0110: fpll-div/128 1XX0: fpll-div/32 0X01: fs/2 0X11: fs/8 1XX1: fs/4

6.2.3 Programmable Timer Registers

Each of Timer 0 to 3 has 8-bit programmable timer registers.
Programmable timer register consists of compare registers and binary counters.
Compare register is 8-bit register which stores values to be compared with binary counter.

- Timer 0, 1, 2, 3 Compare Register
  (TM0OC: 0x03F62, TM1OC: 0x03F63, TM2OC: 0x03F6A, TM3OC: 0x03F6B)

bp	7	6	5	4	3	2	1	0
Flag	TMnOC7	TMnOC6	TMnOC5	TMnOC4	TMnOC3	TMnOC2	TMnOC1	TMnOC0
At reset	X	X	X	X	X	X	X	X
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Binary counter is an 8-bit up counter. If any data is written to compare register when counter is stopped, binary counter is cleared to 0x00.

- Timer 0, 1, 2, 3 Binary Counter
  (TM0BC: 0x03F60, TM1BC: 0x03F61, TM2BC: 0x03F68, TM3BC: 0x03F69)

bp	7	6	5	4	3	2	1	0
Flag	TMnBC7	TMnBC6	TMnBC5	TMnBC4	TMnBC3	TMnBC2	TMnBC1	TMnBC0
At reset	0	0	0	0	0	0	0	0
Access	R	R	R	R	R	R	R	R

6.2.4 Timer Mode Registers

Timer mode register is readable/writable register that controls Timer 0 to 3.

■ Timer 0 Mode Register (TM0MD: 0x03F64)

bp	7	6	5	4	3	2	1	0
Flag	-	TM0POP	TM0MOD	TM0PWM	TM0EN	TM0CK2	TM0CK1	TM0CK0
At reset	-	0	0	0	0	0	0	0
Access	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7	-	-
6	TM0POP	Output signal start polarity selection 0: Timer output Low→High, PWM High→Low 1: Timer output High→Low, PWM Low→High
5	TM0MOD	Pulse width measurement control 0: Normal timer operation 1: P20 pulse width measurement
4	TM0PWM	Timer 0 operation mode selection 0: Normal timer operation 1: PWM operation
3	TM0EN	Timer 0 count control 0: Halt the count 1: Operate the count
2 to 0	TM0CK2 TM0CK1 TM0CK0	Clock source selection X00: fpll_div X01: TM0PSC (prescaler output) 010: fx 011: Synchronous fx 110: TM0IO input 111: Synchronous TM0IO output

To load data from a compare register to an internal register while counting is stopped, one count clock is required. The count clock should be input from external when the timer input is selected as the clock source.

■ Timer 1 Mode Register (TM1MD: 0x03F65)

bp	7	6	5	4	3	2	1	0
Flag	Reserved	Reserved	TM1CAS1	TM1CAS0	TM1EN	TM1CK2	TM1CK1	TM1CK0
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7 to 6	Reserved	Always set to "0".
5 to 4	TM1CAS1 TM1CAS0	Timer 1 operation mode selection 00: Normal operation 01: 16-bit cascade connection 10: 24-bit cascade connection 11: 32-bit cascade connection
3	TM1EN	Timer 1 count control 0: Halt the count 1: Operate the count
2 to 0	TM1CK2 TM1CK1 TM1CK0	Clock source selection X00: fpll_div X01: TM1PSC (prescaler output) 010: fx 011: Synchronous fx 110: TM1IO input 111: Synchronous TM0IO input

To load data from a compare register to an internal register while counting is stopped, one count clock is required. The count clock should be input from external when the timer input is selected as the clock source.

- Timer 2 Mode Register (TM2MD: 0x03F6C)

bp	7	6	5	4	3	2	1	0
Flag	-	TM2POP	TM2MOD	TM2PWM	TM2EN	TM2CK2	TM2CK1	TM2CK0
At reset	-	0	0	0	0	0	0	0
Access	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7	-	-
6	TM2POP	Output signal start polarity selection 0: Timer output Low→High, PWM High→Low 1: Timer output High→Low, PWM Low→High
5	TM2MOD	Pulse width measurement control 0: Normal timer operation 1: P22 pulse width measurement
4	TM2PWM	Timer 2 operation mode selection 0: Normal timer operation 1: PWM operation
3	TM2EN	Timer 2 count control 0: Halt the count 1: Operate the count
2 to 0	TM2CK2 TM2CK1 TM2CK0	Clock source selection X00: fpll_div X01: TM2PSC (prescaler output) 010: fx 011: Synchronous fx 110: TM2IO input 111: Synchronous TM2IO output

To load data from a compare register to an internal register while the counter is stopped, one count clock is required. The count clock should be input externally when the timer input is selected as the clock source.

■ Timer 3 Mode Register (TM3MD: 0x03F6D)

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	TM3CAS	TM3EN	TM3CK2	TM3CK1	TM3CK0
At reset	-	-	-	0	0	0	0	0
Access	-	-	-	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7 to 5	-	-
4	TM3CAS	Timer 3 operation mode selection 0: Normal timer operation 1: 16-bit cascade connection
3	TM3EN	Timer 3 count control 0: Halt the count 1: Operate the count
2 to 0	TM3CK2 TM3CK1 TM3CK0	Clock source selection X00: fpll-div X01: TM3PSC (prescaler output) 010: fx 011: Synchronous fx 110: TM3IO input 111: Synchronous TM3IO input

To load data from a compare register to an internal register while the counter is stopped, one count clock is required. The count clock should be input externally when the timer input is selected as the clock source.

- Timer Clock Selection Register 1 (TMCKSEL1: 0x03FB0)

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	TM3IOSEL	TM2IOSEL	TM1IOSEL	TM0IOSEL
At reset	-	-	-	-	0	0	0	0
Access	-	-	-	-	R/W	R/W	R/W	R/W

bp	Flag	Description
7-4	-	-
3	TM3IOSEL	Timer 3 input selection 0: TM3IO 1: Timer A
2	TM2IOSEL	Timer 2 input selection 0: TM2IO 1: Timer A
1	TM1IOSEL	Timer 1 input selection 0: TM1IO 1: Timer A
0	TM0IOSEL	Timer 0 input selection 0: TM0IO 1: Timer A

- Timer Input Selection Register 1 (TMINSEL1: 0x03FB2)

bp	7	6	5	4	3	2	1	0
Flag	TMINSEL 17	TMINSEL 16	TMINSEL 15	TMINSEL 14	TMINSEL 13	TMINSEL 12	TMINSEL 11	TMINSEL 10
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-6	TMINSEL17 TMINSEL16	Timer 3 input selection (at port selection) 00: TM3IOA 01: TM3IOB 10: “1” is input to timer 11: “1” is input to timer
5-4	TMINSEL15 TMINSEL14	Timer 2 input selection (at port selection) 00: TM2IOA 01: TM2IOB 10: “1” is input to timer 11: “1” is input to timer
3-2	TMINSEL13 TMINSEL12	Timer 1 input selection (at port selection) 00: TM1IOA 01: TM1IOB 10: “1” is input to timer 11: “1” is input to timer
1-0	TMINSEL11 TMINSEL10	Timer 0 input selection (at port selection) 00: TM0IOA 01: TM0IOB 10: “1” is input to timer 11: “1” is input to timer

6.3 Prescaler

6.3.1 Prescaler Operation

■ Prescaler Operation (Prescaler 0 and 1)

Prescaler 0 and Prescaler 1 are 7-bit and 3-bit free-run counter respectively. They output the dividing clock of the reference clock. This count up operation starts automatically when any TMnEN flags of 8-bit timer are set to "1" and operate the Timer n counting. Also, it stops automatically when all TMnEN flags of 8-bit timer are set to "0" and stop all timer counting.

■ Count Timing of Prescaler Operation (Prescaler 0 and 1)

Prescaler 0 counts up at the rising edge of fpll_div.
Prescaler 1 counts up at the rising edge of fs.

■ Peripheral Functions

The table below shows peripheral functions that use prescaler output dividing clock and registers that select dividing clocks.

Table:6.3.1 Prescaler for Peripheral Functions

Timer 0 Count Clock	CK0MD
Timer 1 Count Clock	CK1MD
Timer 2 Count Clock	CK2MD
Timer 3 Count Clock	CK3MD

Start the timer operation after the prescaler setup.
At timer side, the prescaler output should be set up by the timer mode register.
The prescaler starts counting at the start of the timer operation.

6.3.2 Setup Example

■ Prescaler Operation Setup Example

Clock fs/2 which is output from prescaler 1 is selected as the count clock of Timer 0.
The description below is an setup procedure example.

Setup Procedure	Description
(1) Select the prescaler output CK0MD(0x03F66) bp2 to 1: TM0PSC1 to 0 =X0 bp0: TM0BAS =1	(1) Select “fs/2” to the prescaler output by the TM0PSC1 to 0 and TM0BAS flags of CK0MD register.

At timer side, prescaler output selection should be set up by the timer mode register.

6.4 8-bit Timer Count

6.4.1 8-bit Timer Operation

Timers have a function to generate interrupts regularly.

■ 8-bit Timer Operation (Timer 0, 1, 2 and 3)

The generation cycle of timer interrupts is set by the clock source selection and the setting value of TMnOC register, in advance. If TMnBC reaches the setting value of TMnOC register, an interrupt is generated at the next count clock, then TMnBC is cleared and counting restarts from "0x00".

The table below shows clock sources that can be selected by timer.

Table:6.4.1 Clock Source of Timers

Clock source	Time per count	Timer 0 (8-bit)	Timer 1 (8-bit)	Timer 2 (8-bit)	Timer 3 (8-bit)
fpll-div	100 ns	√	√	√	√
fpll-div/4	400 ns	√	√	√	√
fpll-div/16	1.6 µs	√	√	√	√
fpll-div/32	3.2 µs	√	√	√	√
fpll-div/64	6.4 µs	√	√	√	√
fpll-div/128	12.8 µs	√	√	√	√
fs/2	400 ns	√	√	√	√
fs/4	800 ns	√	√	√	√
fs/8	1.6 µs	√	√	√	√
fx	30.5 µs	√	√	√	√

fpll-div = 10 MHz, fx = 32.768 kHz
fs = fpll-div/2 = 5 MHz

When fx is used as clock source, timer counter is counted at hte falling of the count clock and others are used, timer counter is counted at the rising of the count clock.

■ Count Timing of Timer Operation (Timer 0, 1, 2 and 3)

Binary counter counts up with selected clock source as a count clock.
The basic operation of the whole function of 8-bit timer is as follows:

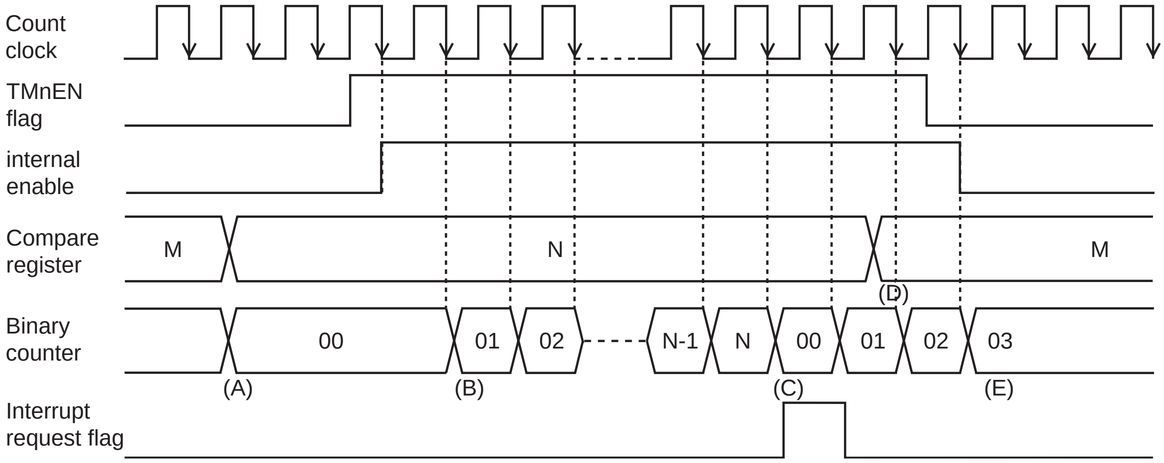


Figure:6.4.1 Count Timing of Timer Operation (Timer 0, 1, 2 and 3)

- (A) If the value is written to compare register when the TMnEN flag is "0", binary counter is cleared to "0x00".
- (B) When the TMnEN flag is "1", the internal enable will be turned on at the next count clock. Then binary counter begins counting up.
- (C) If binary counter reaches the value of compare register, the interrupt request flag is set at the next count clock, then binary counter is cleared to "0x00" and the counting is restarted.
- (D) Even if compare register is rewritten when the TMnEN flag is "1", binary counter is not changed.
- (E) When the TMnEN flag is "0", the internal enable will be turned off at the next count clock. As a result, binary counter stops counting.

Switch the count clock after the timer operation is stopped, since the counting is not generated correctly during the timer operation.

TMnEN flag of the TMnMD and other bits should not be changed at the same time to operate correctly.

When the binary counter reaches the value in the compare register, the interrupt request flag is set and the binary counter is cleared, at the next count clock. So set the compare register as:

Compare register setting = (count till the interrupt request -1)

However, the interrupt generation cycle when the compare register is set to “00” is the same as it is set to “01”.

If the compare register is set the smaller than the binary counter during the count operation, the binary counter counts up to the overflow, at first.

If the timer interrupt request flag may have already been set before timer is started, the timer interrupt request flag should be cleared.

After the timer interrupt request flag is generated, maximum 3 system clock is required until the next interrupt request flag is generated. During this period, the interrupt request flag is not generated even if binary counter reaches the set value of compare register.

When TMnOC register is set to “00”, clear binary counter before starting the operation.

The binary counter of the 8-bit timer is controlled by signals generated by sampling values of the TMnEN flag with the count clock. If the low-speed clock (fx) is selected as a count clock source, it is important to remember the followings.

After the timer halts, read the binary counter as follows:
Wait until another count clock passes, then read the value. In other words, the value is one count less than the actual value if you don’t wait. Another options is that read more than once to obtain accurate value by an appropriate program.

In case of changing the timer settings (selecting clocks or functions, for example) while the timer is suspended, let the count clock count one more after the flag has been turned off. Otherwise the results will not be guaranteed.

If CPU OPERATION mode is changed (from NORMAL to SLOW) when the high-frequency oscillation clock (fpll-div) or the prescaler output (TMnPSC) is selected as clock source, the timer operation should be stopped before operation mode transition and should be reset to start the timer after operation mode transition.
In the SLOW/HALT1 mode, as timer clock source, do not select fpll-div or the clock generated from fpll-div.

If the low-speed clock (fx) is selected as a count clock source, the delay produced by the binary counter may give a wrong value.
Do not write values into the compare register (TMnOC) during counting. Selecting the synchronous low-speed clock (fx) as a count clock source solves those problems; getting correct values and allowing to write into the register during counting.

6.4.2 Setup Example

■ Timer Operation Setup Example (Timer 0,1, 2 and 3)

Timer function can be set by using timer 0 that generates the constant interrupt. Interrupt is generated every 250 cycles (200 μs) by selecting fs/2 (at fs=2.5 MHz operation) as a clock source.

A setup procedure example, with a description of each step is shown below.

Setup Procedure	Description
(1) Stop the counter TM0MD(0x03F64) bp3: TM0EN =0	(1) Set the TM0EN flag of TM0MD register to "0" to stop the counting of Timer 0.
(2) Disable the interrupt TM0ICR(0x03FEF) bp1: TM0IE =0	(2) Set the TM0IE flag of TM0ICR register to "0" to disable the interrupt.
(3) Select the normal timer operation TM0MD(0x03F64) bp4: TM0PWM =0 bp5: TM0MOD =0	(3) Set the TM0PWM flag and the TM0MOD flag of TM0MD register to "0" to select the normal timer operation.
(4) Select the count clock source TM0MD(0x03F64) bp2 to 0: TM0CK2 to 0 =X01	(4) Select the prescaler output to the clock source by the TM0CK2 to 0 flags of TM0MD register.
(5) Select and enable the prescaler output CK0MD(0x03F66) bp3 to 1: TM0PSC2 to 0 =0X0 bp0: TM0BAS =1	(5) Select fs/2 to the prescaler output by the TM0PSC 2 to 0 flags and TM0BAS flag of CK0MD register.
(6) Set the cycle of the interrupt generation TM0OC (0x03F62) =0xF9	(6) Set the value of the interrupt generation cycle to TM0OC register. The cycle is 250, so that the setting value is set to 249 (0xF9). At that time, TM0BC is initialized to 0x00.
(7) Set the interrupt level TM0ICR(0x03FEF) bp7 to 6: TM0LV1 to 0 =10	(7) Set the interrupt level by the TM0LV1 to 0 flags of TM0ICR register. If the interrupt request flag has already been set, clear the request flag. [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup]
(8) Enable the interrupt TM0ICR (0x03FEF) bp1: TM0IE =1	(8) Set the TM0IE flag of TM0ICR register to "1" to enable the interrupt.
(9) Start the timer operation TM0MD(0x03F64) bp3: TM0EN =1	(9) Set the TM0EN flag of TM0MD register to "1" to operate Timer 0.

TM0BC starts to count up from 0x00. When TM0BC reaches the setting value of TM0OC register, Timer 0 interrupt request flag is set at the next count clock, then the value of TM0BC becomes 0x00 and restart to count up.

6.5 8-bit Event Count

6.5.1 Operation

Event count operation has 2 types; TMnIO input and synchronous TMnIO input, according to the clock source selection.

■ 8-bit Event Count Operation (Timer 0, 1, 2 and 3)

Event count operation means that TMnBC counts the input signal from external to TMnIO pin. If the value of binary counter reaches the setting value of TMnOC register, interrupts can be generated at the next count clock.

Table:6.5.1 Event Count Input Clock

Table remarks √: With function -: Without function

	Event input			MN101EFA8	MN101EFA3	MN101EFA7	MN101EFA2
Timer 0	TM0IOA	Synchronous TM0IO input	Timer A output	√	√	√	√
	TM0IOB			√	√	√	√
Timer 1	TM1IOA	Synchronous TM1IO input		√	√	√	√
	TM1IOB			√	√	√	√
Timer 2	TM2IOA	Synchronous TM2IO input		√	√	√	√
	TM2IOB			√	√	√	√
Timer 3	TM3IOA	Synchronous TM3IO input		√	√	√	√
	TM3IOB			√	√	√	√

■ Count Timing of TMnIO Input (Timer 0,1, 2 and 3)

When TMnIO input is selected, TMnIO is input to the count clock of Timer n.
Binary counter is started to count up at the falling edge of the TMnIO input signal.

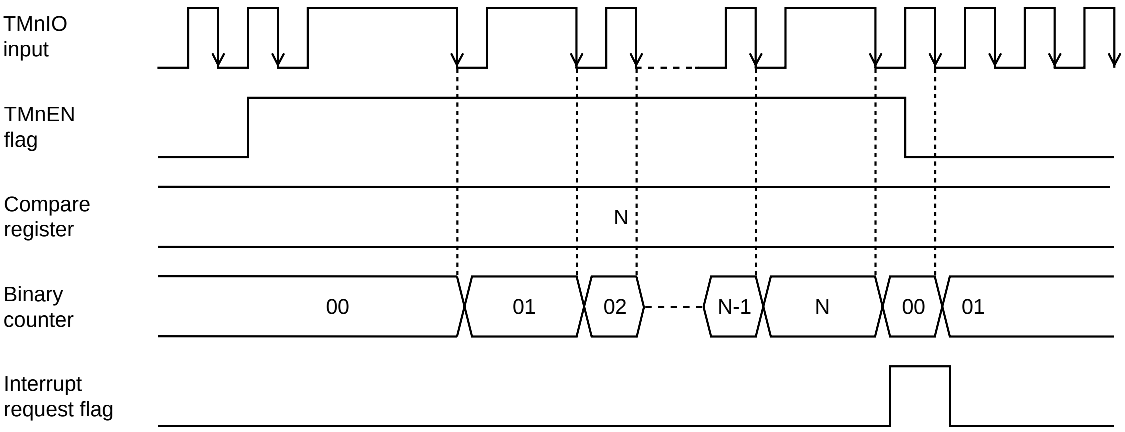


Figure:6.5.1 Count Timing of TMnIO Input (Timer 0, 1, 2 and 3)

If the TMnIO input is selected as a count clock source, the delay produced by the binary counter may give a wrong value.
Do not write values into TMnOC register during counting. Selecting the event count of the synchronous TMnIO input as a count clock source solves those problems; getting correct values and allowing to write into the register during counting.

Binary counter of the 8-bit timer is controlled by signals generated by sampling values of the TMnEN flag with the count clock. If the TMnIO Input is selected as a count clock source, it is important to remember the followings.

After the timer halts, read binary counter as follows:
Wait until another count clock passes, then read the value. In other words, the value is one count less than the actual value if you don’t wait. Another options is that read more than once to obtain accurate value by an appropriate program.

In case of modify the timer settings (selecting clocks or functions, for example) while the timer is suspended, let the count clock count one more after the flag has been turned off. Otherwise the results will not be guaranteed.

When using the event input, clear binary counter before starting timer operation.

Under the following conditions, be sure to select a system clock (fs) as a clock source of timers to set compare register of timer n.

<Conditions applicable to this note>
When one of the following conditions is met while timer n is halted:

	Relevant Block	TMnMD (Timer n mode register)	The frequency of system clock (fs) and high-speed clock for peripheral functions (fpll-div)
		TMnCK2 to 0	
Condition 1	Timer n	X00 (clock source: fpll-div)	The frequency of a system clock (fs) is faster than twice the frequency of a high-speed clock for peripheral functions (fpll-div).

	Relevant Block	TMnMD (Timer n mode register)	CPUM (CPU mode control register)
		TMnCK2 to 0	STOP/HALT/OSC1/OSC0
Condition 2	Timer n	010 (clock source: fslow)	0000 (Normal mode)

	Relevant Block	TMnMD (Timer n mode register)
		TMnCK2 to 0
Condition 3	Timer n	110 (clock source: TMnIO input)

■ Count Timing of Synchronous TMnIO Input (Timer 0, 1, 2 and 3)

When the synchronous TMnIO input is selected, fs is input to Timer n count clock.
The count enable signal is synchronized with the falling edge of system clock.

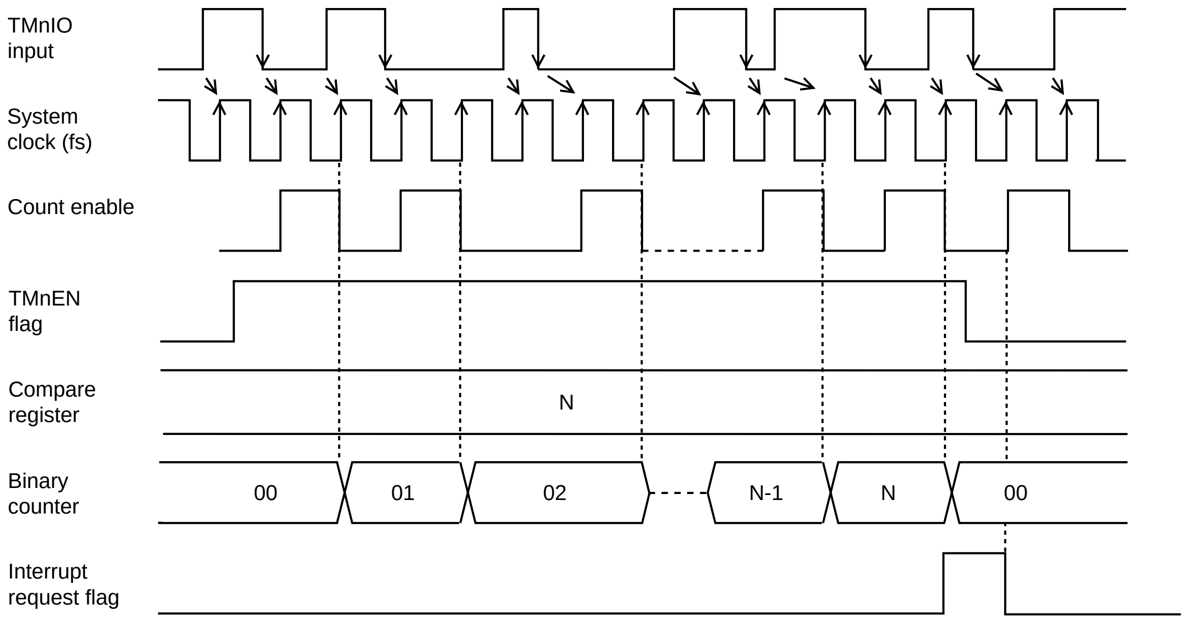


Figure:6.5.2 Count Timing of Synchronous TMnIO Input (Timer 0, 1, 2 and 3)

When the synchronous TMnIO input is selected as the count clock source, Timer n counter counts up in synchronization with system clock, therefore the correct value is always read out.

Input signal from TMnIO should be set with the cycle more than double the system clock (fs). When other signals with a cycle shorter than this are input, the counting may not be performed correctly.

6.5.2 Setup Example

■ Event Count Setup Example (Timer 0, 1, 2 and 3)

When the falling edge of the TM0IO input pin signal is detected for 5 times, an interrupt is generated.
A setup procedure example, with a description of each step is shown below.

Setup Procedure	Description
(1) Stop the counter TM0MD(0x03F64) bp3: TM0EN =0	(1) Set the TM0EN flag of TM0MD register to "0" to stop Timer 0 counting.
(2) Disable the interrupt TM0ICR(0x03FEF) bp1: TM0IE =0	(2) Set the TM0IE flag of TM0ICR register to "0" to disable the interrupt.
(3) Set the special function pin to input PADIR(0x03E9A) bp4: PADIR4 =0 TMINSEL1(0x03FB2) bp1 to 0: TMINSEL11 to 10 =00 TMCKSEL1(0x03FB0) bp0: TM0IOSEL =0	(3) Set the PADIR4 flag of PADIR register to "0", TMINSEL11 to 10 flags of TMINSEL1 register to “00” and TM0IOSEL of TMCKSEL1 register to “0” in order to set PA4 pin to input mode. [Chapter 5 I/O Port]
(4) Set the interrupt generation cycle TM0OC (0x03F62) =0x04	(4) Set the interrupt generation cycle to TM0OC register. As timer executes counting 5 times, the setting value should be 0x04. At the time, TM0BC is initializes to 0x00.
(5) Select the normal timer operation TM0MD(0x03F64) bp4: TM0PWM =0 bp5: TM0MOD =0	(5) Set the TM0PWM flag and the TM0MOD flag of TM0MD register to "0" to select the normal timer operation.
(6) Select the count clock source TM0MD(0x03F64) bp2 to 0: TM0CK2 to 0 =110	(6) Select the TM0IO input to the clock source by the TM0CK2 to 0 flags of TM0MD register.
(7) Set the interrupt level TM0ICR(0x03FEF) bp7 to 6: TM0LV1 to 0 =10	(7) Set the interrupt level by the TM0LV1 to 0 flags of TM0ICR register. If the interrupt request flag has already been set, clear the request flag. [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup]
(8) Enable the interrupt TM0ICR(0x03FEF) bp1: TM0IE =1	(8) Set the TM0IE flag of TM0ICR register to "1" to enable the interrupt.
(9) Start the event count TM0MD(0x03F64) bp3: TM0EN =1	(9) Set the TM0EN flag of TM0MD register to "1" to operate Timer 0.

Every time TM0BC detects the falling edge of TM0IO input, TM0BC counts up from 0x00. When TM0BC reaches the setting value of TM0OC register, Timer 0 interrupt request flag is set at the next count clock, then the value of TM0BC becomes 0x00 and counting up is restarted.

6.6 8-bit Timer Pulse Output

6.6.1 Operation

Pin TMnIO can output a pulse signal at any frequency.

■ Operation of Timer Pulse Output (Timer 0, 1, 2 and 3)

Timer can output signals of 2 × cycle of the setup value in TMnOC register. Output pins are as follows;

Table:6.6.1 Timer Pulse Output Pin

Table remarks √: With function -: Without function

	Pulse output pin	MN101EFA8	MN101EFA3	MN101EFA7	MN101EFA2
Timer 0	TM0IOA output	√	√	√	√
	TM0IOB output	√	√	√	√
Timer 1	TM1IOA output	√	√	√	√
	TM1IOB output	√	√	√	√
Timer 2	TM2IOA output	√	√	√	√
	TM2IOB output	√	√	√	√
Timer 3	TM3IOA output	√	√	√	√
	TM3IOB output	√	√	√	√

■ Count Timing of Timer Pulse Output (Timer 0, 1, 2 and 3)

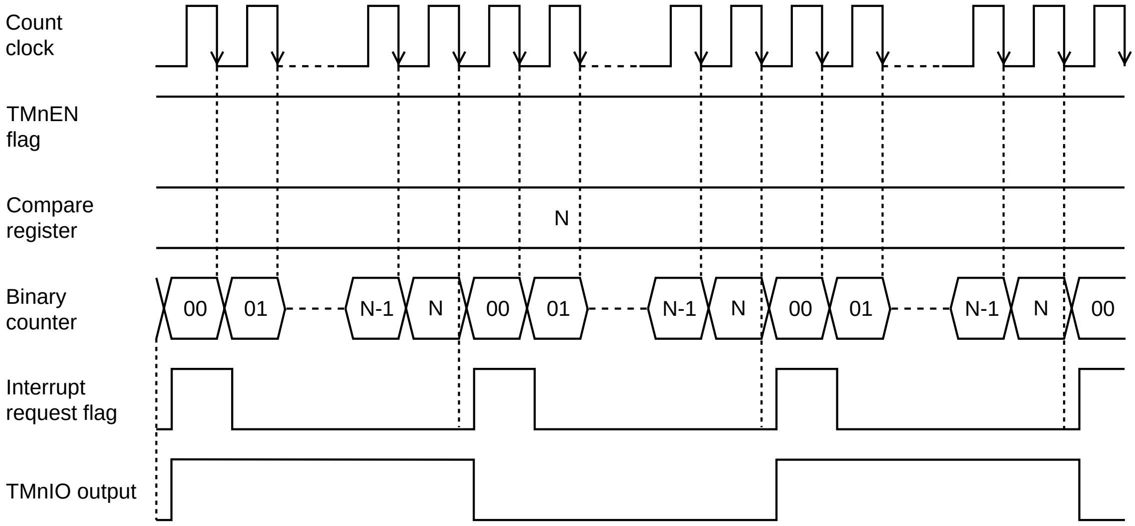


Figure:6.6.1 Count Timing of Timer Pulse Output (Timer 0, 1, 2 and 3)

- Pin TMnIO outputs signals of 2 × cycle of the setup value in compare register. If binary counter reaches compare register, and binary counter is cleared to 0x00, TMnIO output (timer output) is inverted.

When the synchronous TMnIO input is selected as the count clock source, Timer n counter counts up in synchronization with system clock, therefore the correct value is always read out.

6.6.2 Setup Example

■ Timer Pulse Output Setup Example (Timer 0, 1, 2 and 3)

TM0IO pin outputs 50 kHz pulse by using Timer 0. For this, select fs/2 for clock source, and set a 1/2 cycle (100 kHz) for Timer 0 compare register (at fs = 10 MHz).
An example setup procedure, with a description of each step is shown below.

Setup Procedure	Description
(1) Stop the counter TM0MD(0x03F64) bp3: TM0EN =0	(1) Set the TM0EN flag of TM0MD register to "0" to stop Timer 0 counting.
(2) Set the special function pin to the output mode PAOMD(0x03EBA) bp4: PAOMD4 =1 PADIR (0x03E9A) bp4: PADIR4 =1	(2) Set the PAOMD4 flag of PAOMD register to "1" to set PA4 pin to the special function pin. Set the PADIR4 flag of PADIR register to "1" to set the output mode. [Chapter 5 I/O Port]
(3) Select the normal timer operation TM0MD(0x03F64) bp4: TM0PWM =0 bp5: TM0MOD =0	(3) Set the TM0MOD flag of TM0MD register to "0" to select the normal timer operation.
(4) Select the count clock source TM0MD(0x03F64) bp2 to 0: TM0CK2 to 0 =X01	(4) Select the prescaler output to the clock source by the TM0CK2 to 0 flags of TM0MD register.
(5) Select and enable the prescaler output CK0MD(0x03F66) bp3 to 1: TM0PSC2 to 0 =0X0 bp0: TM0BAS =1	(5) Select fs/2 to the prescaler output by the TM0PSC2 to 0 flags and TM0BAS flag of CK0MD register.
(6) Set the timer pulse output cycle TM0OC (0x03F62) =0x31	(6) Set TM0OC register to the 1/2 of the timer pulse output cycle. The setting value should be 50-1=49 (0x31), for 100 kHz to be divided by 5 MHz. At that time, TM0BC is initialized to 0x00.
(7) Start the timer operation TM0MD(0x03F64) bp3: TM0EN =1	(7) Set the TM0EN flag of TM0MD register to "1" to operate Timer 0.

TM0BC counts up from 0x00. If TM0BC reaches the setting value of TM0OC register, then TM0BC is cleared to 0x00, TM0IO output signal is inverted and TM0BC restarts to count up from 0x00.

If any data is written to compare register when binary counter is stopped, the timer output turns to "Low".

[Calculation of compare register value]
Compare register = Timer pulse output / (Selection clock cycle × 2) - 1

6.7 8-bit PWM Output

Pin TMnIO outputs the PWM waveform, which is generated when values of both binary counter and compare register match and when binary counter overflows.

6.7.1 Operation

■ Operation of 8-bit PWM Output (Timer 0 and 2)

The PWM waveform with an arbitrary duty cycle is generated by setting the duty cycle of PWM "High" period to TMnOC register. The cycle is the period from the full count to the overflow of the 8-bit timer.

Table:6.7.1 shows PWM output pins.

Table:6.7.1 Output Pins of PWM Output

Table remarks √: With function -: Without function

	PWM output pin	MN101EFA8	MN101EFA3	MN101EFA7	MN101EFA2
Timer 0	TM0IOA output	√	√	√	√
	TM0IOB output	√	√	√	√
Timer 2	TM2IOA output	√	√	√	√
	TM2IOB output	√	√	√	√

■ Count Timing of PWM Output (at Normal) (Timer 0 and 2)

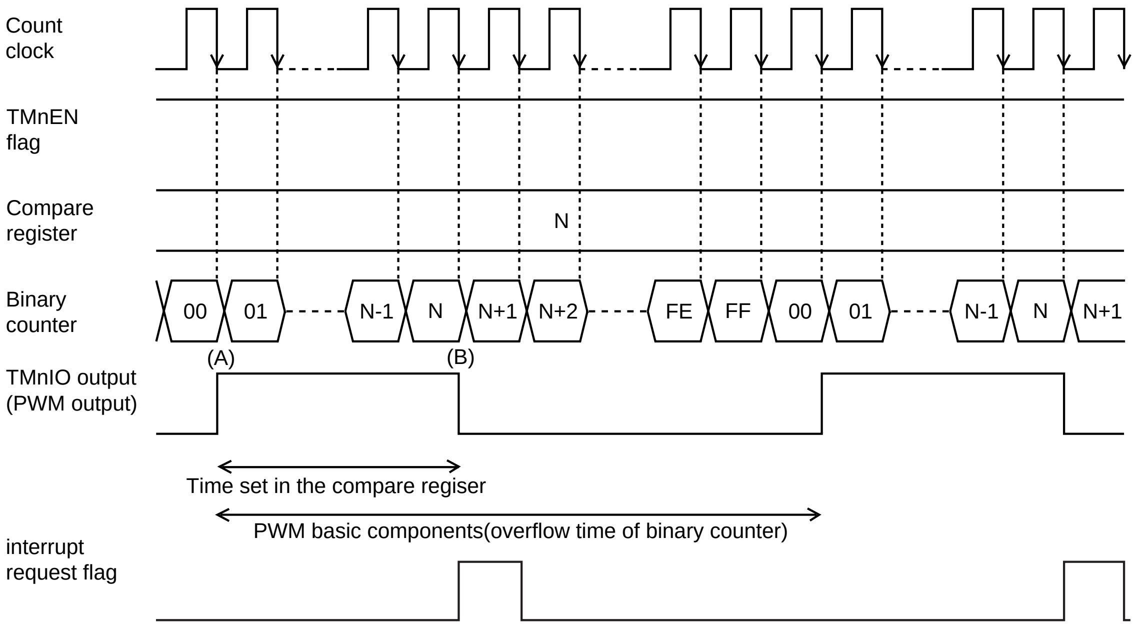

Count clock
TMnEN flag
Compare register
N
Binary counter
00
01
N-1
N
N+1
N+2
FE
FF
00
01
N-1
N
N+1
(A)
(B)
TMnIO output (PWM output)
Time set in the compare regiser
PWM basic components(overflow time of binary counter)
interrupt request flag

Figure:6.7.1 Count Timing of PWM Output (at Normal) (Timer 0 and 2)

When TMnPOP flag is "0":

- (A) PWM output waveform is "High" while binary counter is counting up from 0x01 to the value stored in compare register.
- (B) PWM output waveform is "Low" after the match to the value in compare register, then binary counter continues counting up until it generates an overflow.

The initial setting of PWM output is changed from "Low" to "High" at the selection of PWM operation by the TMnPWM flag of TMnMD register (when TMnPOP flag = 0).

■ Count Timing of PWM Output (when compare register is 0x00) (Timer 0 and 2)

Figure:6.7.2 shows the count timing when compare register is set to 0x00.

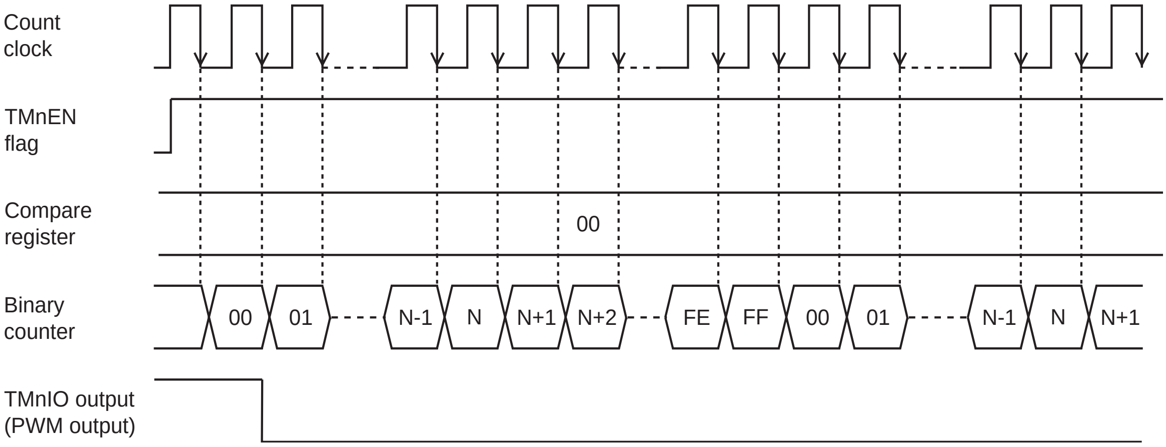


Figure:6.7.2 Count Timing of PWM Output (when compare register is 0x00) (Timer 0 and 2)

When TMnEN flag is stopped ("0"), PWM output is "High".

■ Count Timing of PWM Output (when compare register is 0xFF) (Timer 0 and 2)

Figure:6.7.3 shows the count timing when compare register is set to 0xFF.

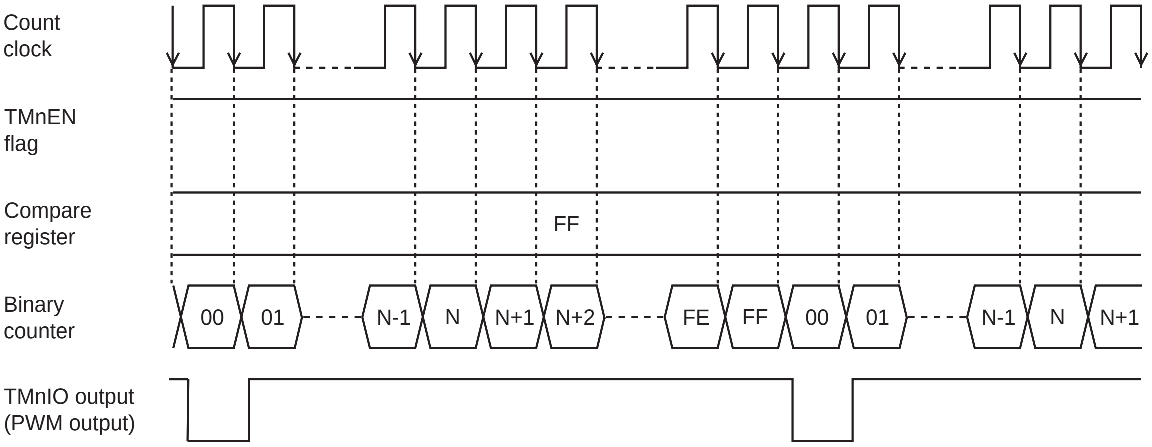


Figure:6.7.3 Count Timing of PWM Output (when compare register is 0xFF) (Timer 0 and 2)

6.7.2 Setup Example

■ PWM Output Setup Example (Timer 0 and 2)

The 1/4 duty cycle PWM output waveform is output from TM0IO output pin at 19.53 Hz by using Timer 0. Clock fs/2 oscillates at 5 MHz. Cycle period of PWM output waveform is decided by the overflow of binary counter. "High" period of the PWM output waveform is decided by the setting value of compare register.

An example setup procedure, with a description of each step is shown below.

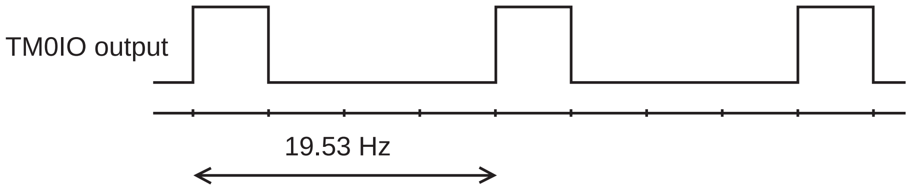


Figure:6.7.4 Output Waveform of TM0IO Output Pin

Setup Procedure	Description
(1) Stop the counter TM0MD(0x03F64) bp3: TM0EN =0	(1) Set the TM0EN flag of TM0MD register to "0" to stop Timer 0 counting.
(2) Set the special function pin to the output mode PAOMD(0x03EBA) bp4: PAOMD4 =1 PADIR (0x03E9A) bp4: PADIR4 =1	(2) Set the PAOMD4 flag of PAOMD register to "1" to set PA4 pin to the special function pin. Set the PADIR4 flag of PADIR register to "1" to set the output mode. [Chapter 5 I/O Port]
(3) Select the PWM operation TM0MD(0x03F64) bp4: TM0PWM =1 bp5: TM0MOD =0 bp6: TM0POP =0	(3) Set the TM0PWM flag of TM0MD register to "1" and the TM0MOD flag to "0" to select the PWM operation.
(4) Select the count clock source TM0MD(0x03F64) bp2 to 0: TM0CK2 to 0 =X01	(4) Select the prescaler output to the clock source by the TM0CK2 to 0 flags of TM0MD register.
(5) Select and enable the prescaler output CK0MD(0x03F66) bp3 to 1: TM0PSC2 to 0 =0X0 bp0: TM0BAS =1	(5) Select fs/2 to the prescaler output by the TM0PSC2 to 0 and TM0BAS flags of CK0MD register.
(6) Set the period of PWM "High" output TM0OC (0x03F62) =0x40	(6) Set the "High" period of PWM output to TM0OC register. The setting value is set to 256/4=64 (0x40), because it should be the 1/4 duty of the full count (256). At that time, TM0BC is initialized to 0x00.
(7) Start the timer operation TM0MD(0x03F64) bp3: TM0EN =1	(7) Set the TM0EN flag of TM0MD register to "1" to operate Timer 0.

6.7.3 PWM Outputs With Additional Pulses

- PWM Output with Additional Pulses Method (Timer 0 and 2)

This method allows the user to add one bit of pulse at a time into a PWM waveform. Up to 3 bits can be added in 4 cycles of the basic waveform.

To determine where to place, or not to place additional bits in the cycles is controlled by CK0MD register or CK2MD register.

- How to add pulses

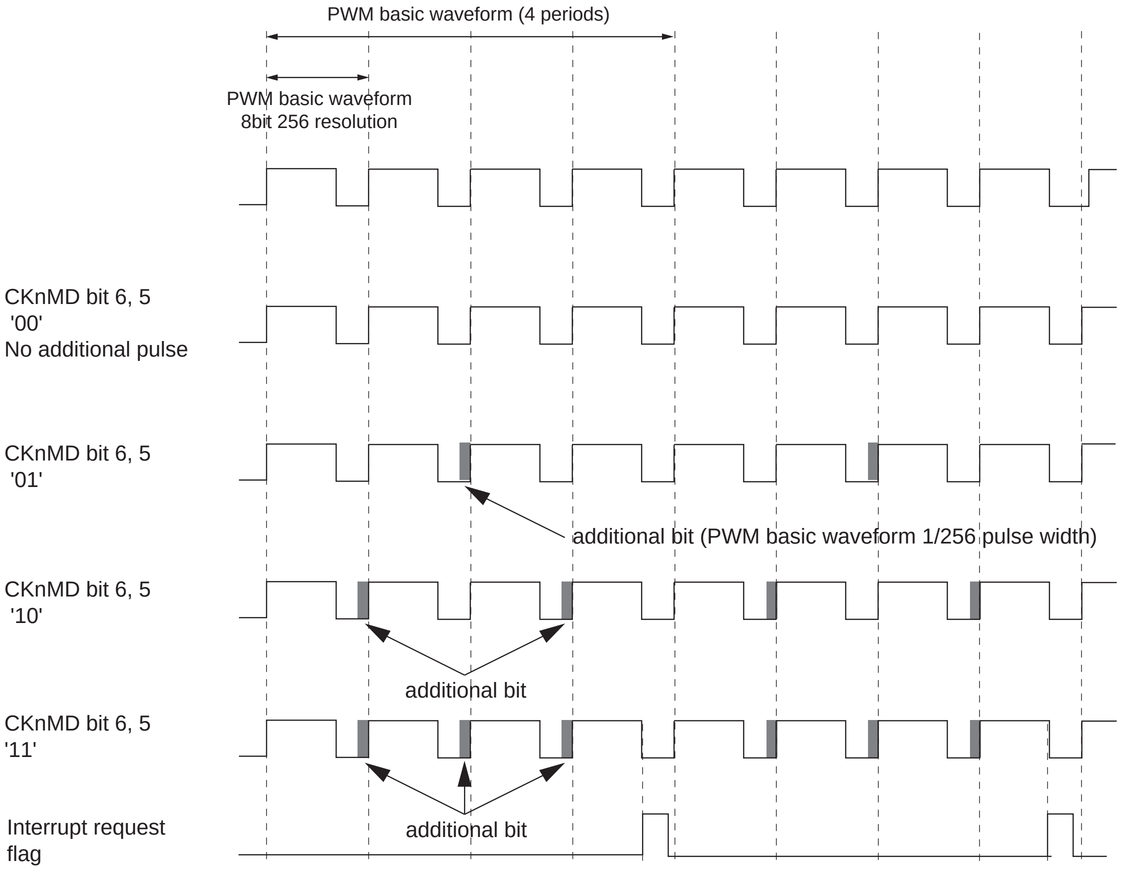
Bit 6 and 5 of CK0MD register or CK2MD register control positions of pulses. For example, if "00" is specified for CK0MD or CK2MD register, not a single pulse will be added to the PWM. However, if "11" is specified, total of 3 bits will be added in the 4 cycles of the PWM.

Table:6.7.2 shows the locations of additional pulses with the values of CK0MD/CK2MD register.
Figure:6.7.5 shows the additional pulses and the PWM waveform.

Table:6.7.2 Additional pulses settings

CKnMD Register		Locations of additional pulses (Within the four cycles of the basic waveform)
bit 6	bit 5	
0	0	No pulse
0	1	At second cycle
1	0	At first and third cycles
1	1	At first, second and third cycles

An interrupt occurs at the 4th cycle of the PWM basic waveform.



During 4 cycles of the PWM basic waveform, additional pulses(1/256 pulse width of PWM basic waveform) can be added in any of the periods 0 to 3.

Figure:6.7.5 Additional pulses and the PWM waveform

6.8 Serial Transfer Clock Output

6.8.1 Operation

Serial transfer clock can be created by using the timer output signal.

Serial transfer clock operation by 8-bit timer (Timer 0, 1, 2 and 3)

- Timer 0: Serial Interface 0, 1, 2, 4
- Timer 1: Serial Interface 0, 1, 2, 4
- Timer 2: Serial Interface 0, 1, 2, 4
- Timer 3: Serial Interface 0, 1, 2, 4

■ Timing of Serial Transfer Clock (Timer 0, 1, 2 and 3)

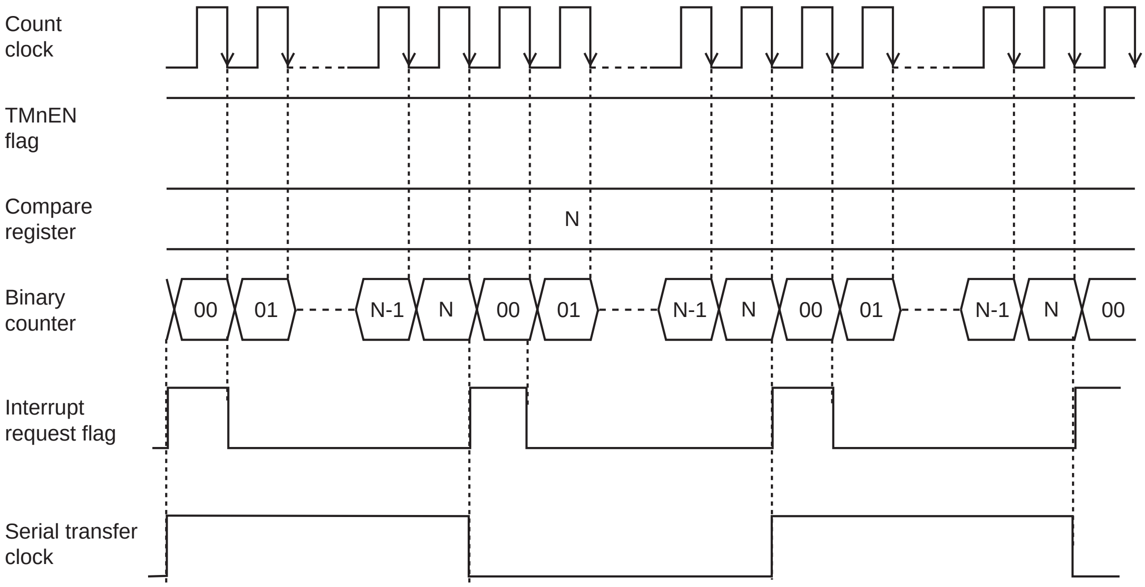


Figure:6.8.1 Timing of Serial Transfer Clock (Timer 0, 1, 2 and 3)

- The timer frequency is 1/2 of the set frequency set by compare register.
- For the baud rate calculation and the serial interface setup, refer to [Chapter 13 Serial Interface].

6.8.2 Setup Example

■ Serial Transfer Clock Setup Example (Timer 0)

Transfer clock for full duplex UART (Serial 0) is generated by using Timer 0.
The baud rate is selected to be 300 bps and the source clock of Timer 0 is selected to be fs/2 (at fs=2 MHz).

An example setup procedure, with a description of each step is shown below.

Setup Procedure	Description
(1) Stop the counter TM0MD(0x03F64) bp3: TM0EN =0	(1) Set the TM0EN flag of TM0MD register to "0" to stop Timer 0 counting.
(2) Select the normal timer operation TM0MD(0x03F64) bp4: TM0PWM =0 bp5: TM0MOD =0	(2) Set the TM0PWM flag and the TM0MOD flag of TM0MD register to "0" to select the normal timer operation.
(3) Select the count clock source TM0MD(0x03F64) bp2 to 0: TM0CK2 to 0 =X01	(3) Select the prescaler output to the clock source by the TM0CK2 to 0 flags of TM0MD register.
(4) Select and enable the prescaler output CK0MD(0x03F66) bp3 to 1: TM0PSC2 to 0 =0X0 bp0: TM0BAS =1	(4) Select fs/2 to the prescaler output by the TM0PSC2 to 0 flags and theTM0BAS flag of CK0MD register.
(5) Set the baud rate TM0OC (0x03F62) =0xCF	(5) Set TM0OC register such a value that the baud rate comes to 300 bps. At that time, TM0BC is initialized to 0x00.
(6) Start the timer operation TM0MD(0x03F64) bp3: TM0EN =1	(6) Set the TM0EN flag of TM0MD register to "1" to operate Timer 0.

- TM0BC counts up from 0x00. Timer 0 output is the clock of Serial Interface 0 at transmission and reception.
- For the setup value of compare register and the setup of the serial interface operation, refer to [Chapter 13 Serial Interface].

6.9 Simple Pulse Width Measurement

6.9.1 Operation

This function is used to measure pulse width during "Low" period of the pulse signal input from the external interrupt pin.

■ Simple Pulse Width Measurement Operation by 8-bit Timer (Timer 0 and 2)

Sample input signals of the external interrupt pins, which is used to measure the simple pulse width, by the count clock. Binary counter will count while the signals are "Low". Pulse width "Low" period can be measured by reading the count of timer. 8-bit timers that have the simple pulse width measurement function are Timer 0 and 2.

Table:6.9.1 Simple Pulse Width Measurable Pins

Table remarks √: With function -: Without function

	Pins that can measure simple pulse width	MN101 EFA8	MN101 EFA3	MN101 EFA7	MN101 EFA2
Timer 0	External Interrupt 0 (P20/IRQ0)	√	√	√	√
Timer 2	External Interrupt 2 (P22/IRQ2)	√	√	√	√

■ Count Timing of Simple Pulse Width Measurement (Timer 0 and 2)

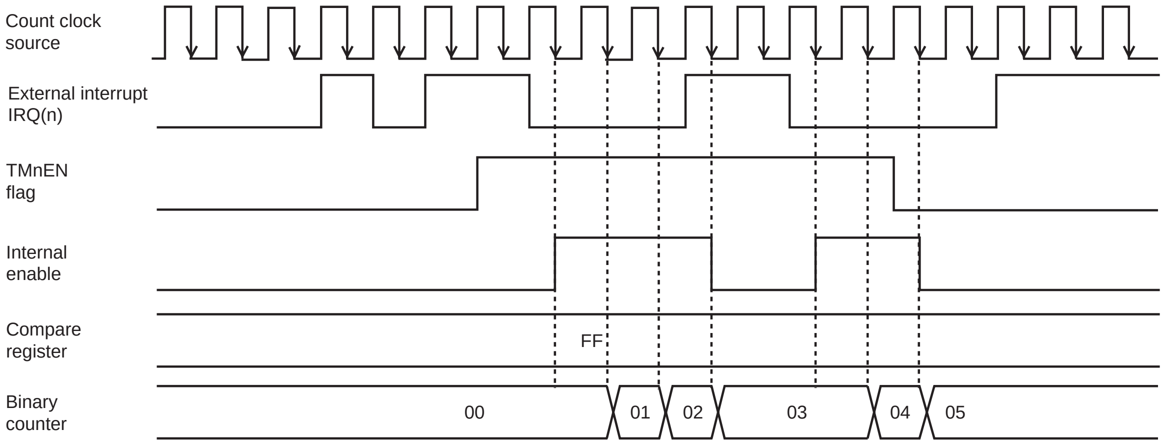


Figure:6.9.1 Count Timing of Simple Pulse Width Measurement (Timer 0 and 2)

- Internal enable signal is generated when the count clock executes samplings of the condition while the TMnEN flag is enable ("1") and the input signal of external interrupt pins for simple pulse width measurement is "Low". While the internal enable signal is "High", timer counts up.

6.9.2 Setup Example

■ Setup Example of Simple Pulse Width Measurement by 8-bit Timer (Timer 0 and 2)

Timer 0 measures pulse width of signals of external interrupt 0 (IRQ0) during "Low" period. The clock source of Timer 0 is selected to fs/2. A setup procedure example, with a description of each step is shown below.

Setup Procedure	Description
(1) Stop the counter TM0MD(0x03F64) bp3: TM0EN =0	(1) Set the TM0EN flag of TM0MD register to "0" to stop Timer 0 counting.
(2) Set the pulse width measurement operation TM0MD(0x03F64) bp4: TM0PWM =0 bp5: TM0MOD =1	(2) Set the TM0PWM flag of TM0MD register to "0" and TM0MOD flag to "1" to enable the timer operation during "Low" period to be measured.
(3) Select the count clock source TM0MD(0x03F64) bp2 to 0: TM0CK2 to 0 =X01	(3) Select the prescaler output to the clock source by the TM0CK2 to 0 flags of TM0MD register.
(4) Select and enable the prescaler output CK0MD(0x03F66) bp3 to 1: TM0PSC2 to 0 =0X0 bp0: TM0BAS =1	(4) Select fs/2 to the prescaler output by the TM0PSC2 to 0 flags and the TM0BAS flag of CK0MD register.
(5) Set the compare register TM0OC (0x03F62) =0xFF	(5) Set TM0OC register to larger value than the cycle of fs/2 / "Low" period of measured pulse width. At that time, TM0BC is initialized to 0x00.
(6) Set the external interrupt IRQCNT (0x03FD0) bp0: P20EN =1	(6) Set the P20EN flag of IRQCNT register to “1” to set P20 as the external interrupt 0 (IRQ0).
(7) Set the interrupt level IRQ0ICR(0x03FE2) bp7 to 6: IRQ0LV1 to 0 =XX	(7) Set the interrupt level by the IRQ0LV1 to 0 flags of IRQ0ICR register. If the interrupt request flag has already been set, clear all interrupt request flags. [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup]
(8) Set the interrupt valid edge IRQ0ICR(0x03FE2) bp5: REDG0 =1	(8) Set the REDG0 flag of IRQ0ICR register to "1" to specify the interrupt valid edge to the rising edge.
(9) Enable the interrupt IRQ0ICR(0x03FE2) bp1: IRQ0IE =1	(9) Set the IRQ0IE flag of IRQ0ICR register to "1" to enable the interrupt.
(10) Enable the timer operation TM0MD(0x03F64) bp3: TM0EN =1	(10) Set the TM0EN flag of TM0MD register to "1" to enable Timer 0 operation.

- The internal enable is set by sampling "Low" level of IRQ0 input with the count clock.
  TM0BC starts counting up from 0x00 after the internal enable is set. Timer 0 continues counting up while IRQ0 is "Low", and the counting is stopped after sampling "High" level of IRQ0. At the same time, reading the value of TM0BC by an interrupt processing can detect "Low" period of IRQ.

Enable the external interrupt input by IRQCNT register when the external interrupt pin is used for the pulse width measurement.

6.10 Cascade Connection

6.10.1 Operation

Cascading Timer 0 and 1 or Timer 2 and 3 forms a 16-bit timer.

■ 16-bit Timer Cascade Connection Operation (Timer 0 + Timer 1, Timer 2 + Timer 3)

Timer 0 and 1 or Timer 2 and 3 are combined to be a 16-bit timer.
Cascading timer is operated at the clock source of Timer 0 which are lower 8 bits.

Table:6.10.1 Timer Functions at Cascade Connection

	Timer 0 + Timer 1 (16bit)	Timer 2 + Timer 3 (16bit)
Interrupt source	TM1IRQ	TM3IRQ
Timer operation	√	√
Event count	TM0IO input	TM2IO input
Timer pulse output	TM1IO output	TM3IO output
PWM output	-	-
Synchronous output	-	-
Pulse width measurement	√	√
Clock source	fpll-div	fpll-div
	fpll-div/4	fpll-div/4
	fpll-div/16	fpll-div/16
	fpll-div/32	fpll-div/32
	fpll-div/64	fpll-div/64
	fpll-div/128	fpll-div/128
	fs/2	fs/2
	fs/4	fs/4
	fs/8	fs/8
	fx	fx
	TM0IO input	TM2IO input
	Synchronous fx	Synchronous fx
	Synchronous TM0IO input	Synchronous TM2IO input
	Timer A input	Timer A input

fpll-div: Machine clock (High-speed oscillation for peripheral functions)
fx: Machine clock (Low-speed oscillation)
fs: System clock

- At cascade connection, the binary counter and the compare register are operated as a 16-bit register.
  At operation, set the TMnEN flag of the lower 8-bit timer to "1" to be operated.

The upper 8-bit timer outputs waveforms of the timer pulse output and interrupt request flags.
Also, select the clock source by the lower 8-bit timer.

Other setup and count timing is the same to the 8-bit timer at independently operation.

When Timer 0 and Timer 1 are used in cascade connection, Timer 1 is used for a timer pulse output and an interrupt request flag. Timer pulse output of Timer 0 is "Low" fixed output.
An interrupt request of Timer 0 is not generated, but Timer 0 interrupt should be disabled.

When Timer 2 and Timer 3 are used in cascade connection, Timer 3 is used for a timer pulse output and an interrupt request flag. Timer pulse output of Timer 2 is "Low" fixed output.
An interrupt request of Timer 2 is not generated, but Timer 2 interrupt should be disabled.

At 16-bit cascade connection, when the clear of binary counter is needed by rewriting compare register, set the TMnEN flag of both the upper 8-bit timer and the lower 8-bit timer to "0" to stop counting. Then rewrite all compare registers.

Use a 16-bit access instruction to set (TM1OC + TM0OC) register and (TM2OC + TM3OC) register.

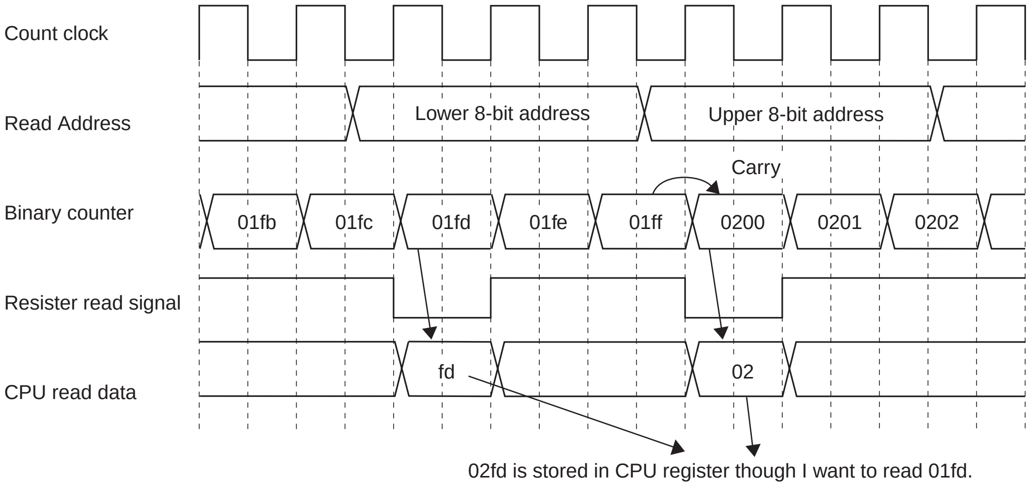
During cascade connection, PWM output function cannot be used. When connecting cascade, always set TMPWM flag of timer mode register to "0".

In reading out value of binary counter during timer operation with 8-bit timer in cascade connection, the value of binary counter may not be correctly read out.

The extended binary counter can not read all bits at a time when 8-bit timer in cascade connection is used as a 16-bit, or more (16, 24 and 32-bit setting are selectable depending on models). Inside LSI, the lower 8-bit is read out separately from the upper 8-bit, even if MOVW instruction of 16-bit access is used.

Therefore, when the value of binary counter is read out in timer operation, the correct value can not be read if a carry from lower 8-bit to upper 8-bit is generated during read out.

Image chart when value of binary counter is read
by two 8-bit accesses with 16-bit timer that connects 8-bit timer in cascade.



Stop the timer in order to read out the correct value of the timer in cascade connection.

6.10.2 24-bit Cascade Connection Operation

Timer 0, 1 and 2 can be cascaded to form a 24-bit timer in 24-bit cascade mode.

- Operation (Timer 0 + Timer 1 + Timer 2)

Timer 0, 1 and 2 can be cascaded to form a 24-bit timer. Then, the timer runs by clock sources of Timer 0 for the lower 8-bit timer.

Table:6.10.2 Timer Functions in Cascade Mode

	Timer 0 + Timer 1 + Timer 2 (24-bit)
Interrupt source	TM2IRQ
Timer operation	√
Event count	TM0IO input
Timer pulse output	TM2IO output
PWM output	-
Synchronous output	-
Pulse width measurement	√
Clock sources	fpll-div
	fpll-div/4
	fpll-div/16
	fpll-div/32
	fpll-div/64
	fpll-div/128
	fs/2
	fs/4
	fs/8
	fx
	TM0IO input
	Synchronous fx
	Synchronous TM0IO input

fpll-div: Machine clock (High speed oscillation for peripheral functions)
fx: Machine clock (Low speed oscillation)
fs: System clock

- At cascade connection, the binary counter and compare register are operated as 24-bit registers. At operation, set the TM0EN flag of Timer 0 to "1".

  Timer 2 outputs waveforms of the timer pulse output and interrupt request flags. Also, select the clock source by Timer 0.

  Other setup and count timing are the same as the settings of 8-bit timer at independently operation.

When Timer 0, 1 and 2 are used in cascade connection, Timer 2 is used for a timer pulse output and an interrupt request flag. Timer pulse output of Timer 0 and 1 is fixed to "Low".
An interrupt request of Timer 0 or Timer 1 is not generated, but Timer 0 and Timer 1 interrupts should be disabled.

At 24-bit cascade connection, when the clear of binary counter is needed by rewriting compare registers, set the TM0EN flag of TM0MD register to "0" to stop counting. Then rewrite all compare registers for Timer 0 to 2.

During cascade connection, PWM output function cannot be used. When connecting cascade, always set TMPWM flag of timer mode register to "0".

Stop the timer in order to read out the correct value of the timer in cascade connection.

6.10.3 32-bit Cascade Connection Operation

Timer 0, 1, 2 and 3 can be cascaded to form a 32-bit timer in 32-bit cascade mode.

■ Operation (Timer 0 + Timer 1 + Timer 2 + Timer 3)

Timer 0, 1, 2 and 3 can be cascaded to form a 32-bit timer. Then, the timer runs by clock sources of Timer 0 for the lower 8-bit timer.

Table:6.10.3 Timer Functions in Cascade Mode

	Timer 0 + Timer 1 + Timer 2 + Timer 3 (32-bit)
Interrupt source	TM3IRQ
Timer operation	√
Event count	TM0IO input
Timer pulse output	TM3IO output
PWM output	-
Synchronous output	√
Pulse width measurement	√
Clock sources	fpll_div
	fpll_div/4
	fpll_div/16
	fpll_div/32
	fpll_div/64
	fpll_div/128
	fs/2
	fs/4
	fs/8
	fx
	TM0IO input
	Synchronous fx
	Synchronous TM0IO input

fpll_div: Machine clock (High speed oscillation for peripheral functions)
fx: Machine clock (Low speed oscillation for peripheral functions)
fs: System clock

- At cascade connection, the binary counter and compare register are operated as 32-bit registers.
  At operation, set the TM0EN flag of Timer 0 to "1".

  Timer 3 outputs waveforms of the timer pulse output and interrupt request flags.
  Also, select the clock source by Timer 0.

  Other setup and count timing are the same as the settings of 8-bit timer at independently operation.

When Timer 0, 1, 2 and 3 are used in cascade connection, Timer 3 is used for a timer pulse output and an interrupt request flag. Timer pulse output of Timer 0, 1 and 2 is fixed to "Low". An interrupt request of Timer 0, 1 or 2 is not generated, but Timer 0, 1 and 2 interrupts should be disabled.

At 32-bit cascade connection, when the clear of the binary counter is needed by rewriting the compare registers, set the TM0EN flag of TM0MD register to "0" to stop counting. Then rewrite all compare registers for Timer 0 to 3.

During cascade connection, PWM output function cannot be used. When connecting cascade, always set TMPWM flag of timer mode register to "0".

At 32-bit cascade connection, always set TM3CAS flag of TM3MD register to "1".

Stop the timer in order to read out the correct value of the timer in cascade connection.

6.10.4 Setup Example

■ Cascade Connection Timer Setup Example (Timer 0 + Timer 1)

Setting example of timer function that an interrupt is constantly generated by cascade connection of Timer 0 and Timer 1, as a 16-bit timer is shown. An interrupt is generated 2500 times every 1 ms by selecting source clock fs/2 (fs=5 MHz at operation). An example setup procedure, with a description of each step is shown below.

Setup Procedure	Description
(1) Stop the counter TM0MD(0x03F64) bp3: TM0EN =0 TM1MD(0x03F65) bp3: TM1EN =0	(1) Set the TM0EN flag of TM0MD register to "0" and the TM1EN flag of TM1MD register to "0" to stop Timer 0 and Timer 1 counting.
(2) Disable the timer interrupt TM0ICR(0x03FEF) bp1: TM0IE =0 TM1ICR(0x03FEA) bp1: TM1IE =0	(2) Set the TM0IE flag of TM0ICR register to "0" to disable the interrupt. Set the TM1IE flag of TM1ICR register to "0" to disable the interrupt.
(3) Select the normal lower timer operation TM0MD(0x03F64) bp4: TM0PWM =0 bp5: TM0MOD =0	(3) Set the TM0PWM flag and the TM0MOD flag of TM0MD register to "0" to select the normal operation mode.
(4) Set the cascade connection TM1MD(0x03F65) bp4: TM1CAS =1	(4) Set the TM1CAS flag of TM1MD register to "0" to connect Timer 1 and Timer 0 to the cascade.
(5) Select the count clock source TM0MD(0x03F64) bp2 to 0: TM0CK2 to 0 =X01	(5) Select the prescaler to the clock source by the TM0CK2 to 0 flags of TM0MD register.
(6) Select and enable the prescaler output CK0MD(0x03F66) bp3 to 1: TM0PSC2 to 0 =0X0 bp0: TM0BAS =1	(6) Select fs/2 to the prescaler output by the TM0PSC2 to 0 flags and the TM0BAS flag of CK0MD register.
(7) Set the interrupt generation cycle TM1OC, TM0OC(0x03F63, 0x03F62) =0x09C3	(7) Set TM1OC and TM0OC registers to the interrupt generation cycle (0x09C3: 2500 cycles -1). At that time, TM1BC and TM0BC are initialized to 0x0000.
(8) Set the level of the upper timer interrupt TM1ICR(0x03FF0) bp7 to 6: TM1LV1 to 0 =10	(8) Set the interrupt level by the TM1LV1 to 0 flags of TM1ICR register. If any interrupt request flag has already been set, clear all request flags. [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup]
(9) Enable the lower timer interrupt TM1ICR(0x03FF0) bp1: TM1IE =1	(9) Set the TM1IE flag of TM1ICR register to "1" to enable the interrupt.
(10) Start the lower timer operation TM0MD(0x03F64) bp3: TM0EN =1	(10) Set the TM0EN flag of TM0MD register to "1" to operate the 16-bit cascade connection.

- TM1BC + TM0BC counts up from 0x0000 as a 16-bit timer.
  When TM1BC + TM0BC reaches the set value of TM1OC + TM0OC register, Timer 1 interrupt request flag is set at the next count clock, and the value of TM1BC + TM0BC becomes 0x0000 and restarts count up.

Chapter 7　8-bit Simple-Timer

7.1 Overview

This timer is a 8-bit simple timer that can be used as a serial transfer clock or timer count clock.

8-bit simple timer is equipped with two prescalers. Each prescaler counts fpll-div and fs as the base clock. Configurations of hardware are shown below.

Prescaler 0 (based on fpll-div)	5-bit Prescaler
Prescaler 1 (based on fs)	2-bit Prescaler

Prescaler 0 outputs fpll-div/2, fpll-div/4, fpll-div/8, fpll-div/16 and fpll-div/32.
Prescaler 1 outputs fs/2 and fs/4.

TMAMD1 register can select a clock source for timer from fpll-div and fs by using the internal prescaler output.

7.1.1 Functions

Table:7.1.1 shows functions that can be used with each timer.

Table:7.1.1 8-bit Simple Timer Functions

	Timer A (8-bit)
Clock source	fpll-div
	fpll-div/2
	fpll-div/4
	fpll-div/8
	fpll-div/16
	fpll-div/32
	fs/2
	fs/4

fpll-div:Machine clock (High speed oscillation for peripheral functions)
fs:System clock

When changing the frequency of fpll-div by bp7 to bp4 of OSCCNT register, it should be executed after the 8-bit simple timer function is stopped.

7.1.2 Block Diagram

■ Timer A Block Diagram

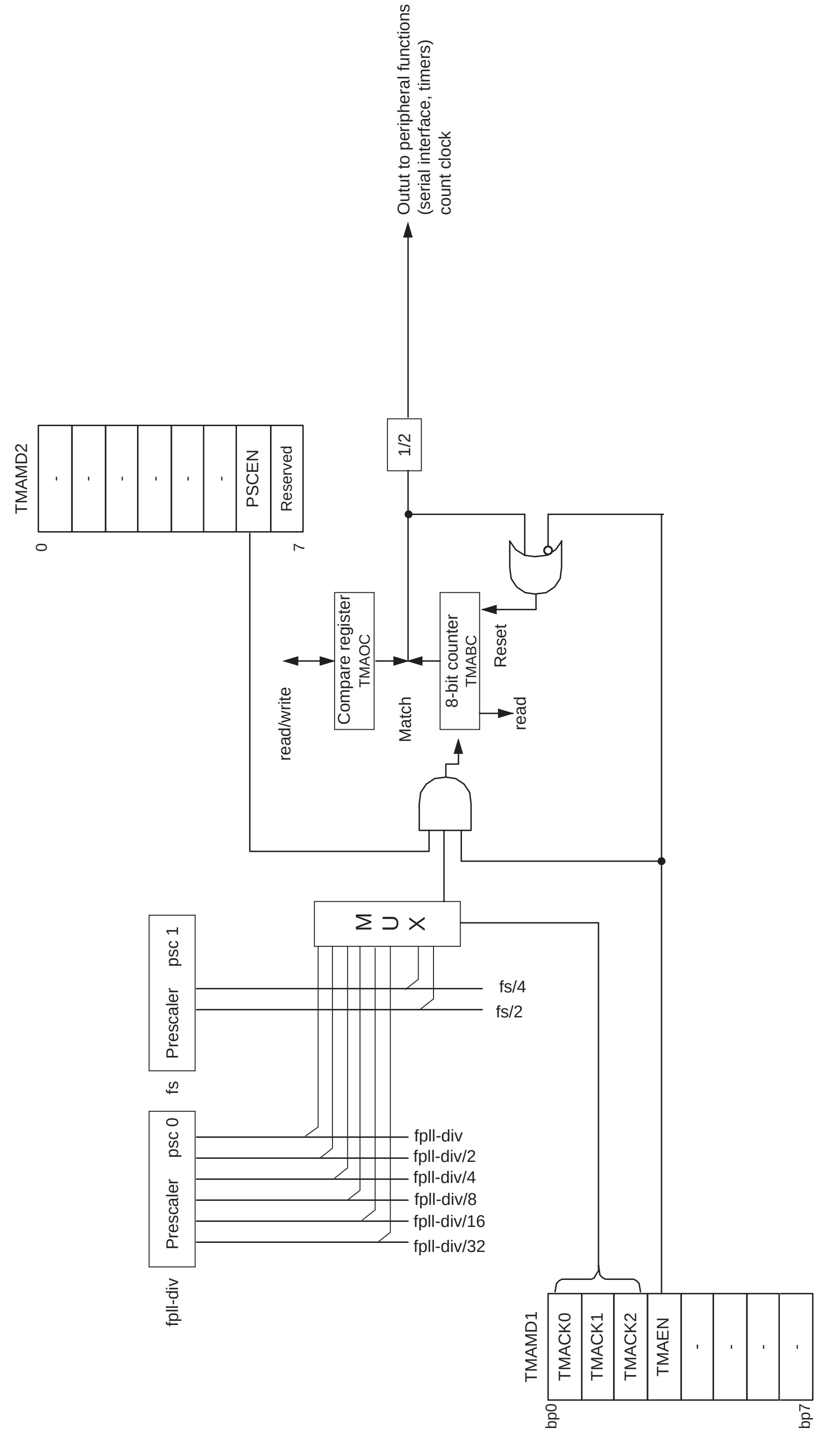


Figure:7.1.1 Timer A Block Diagram

7.2 Control Registers

Timer A consists of binary counter (TMABC) and compare register (TMAOC). It is controlled by mode registers (TMAMD1 and TMAMD2).

7.2.1 Registers

Table:7.2.1 shows registers that control Timer A.

Table:7.2.1 8-bit Timer Control Registers

Table remarks √: With function -: Without function

	Register	Address	R/W	Function	Page	MN101 EFA8	MN101 EFA3	MN101 EFA7	MN101 EFA2
Timer A	TMABC	0x03F71	R	Timer A binary counter	VII-5	√	√	√	√
	TMAOC	0x03F73	R/W	Timer A compare register	VII-5	√	√	√	√
	TMAMD1	0x03F75	R/W	Timer A mode register 1	VII-6	√	√	√	√
	TMAMD2	0x03F77	R/W	Timer A mode register 2	VII-6	√	√	√	√

R/W: Readable/Writable
R: Read only

7.2.2 Programmable Timer Registers

Timer A has 8-bit programmable timer registers.
Programmable timer registers consist of a compare register and a binary counter.
Compare register is a 8-bit register which stores the value to be compared to binary counter.

- Timer A Compare Register (TMAOC: 0x03F73)

bp	7	6	5	4	3	2	1	0
Flag	TMAOC7	TMAOC6	TMAOC5	TMAOC4	TMAOC3	TMAOC2	TMAOC1	TMAOC0
At reset	X	X	X	X	X	X	X	X
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Binary counter is 8-bit up counter. If any data is written to compare register when the counting is stopped, binary counter is cleared to 0x00.

- Timer A Binary Counter (TMABC: 0x03F71)

bp	7	6	5	4	3	2	1	0
Flag	TMABC7	TMABC6	TMABC5	TMABC4	TMABC3	TMABC2	TMABC1	TMABC0
At reset	0	0	0	0	0	0	0	0
Access	R	R	R	R	R	R	R	R

7.2.3 Timer Mode Registers

Timer mode registers are readable/writable registers that control timer A.

- Timer A Mode Register 1 (TMAMD1: 0x03F75)

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	TMAEN	TMACK2	TMACK1	TMACK0
At reset	-	-	-	-	0	0	0	0
Access	-	-	-	-	R/W	R/W	R/W	R/W

bp	Flag	Description
7 to 4	-	-
3	TMAEN	Timer A count control 0: Halt the count 1: Start the count
2 to 0	TMACK2 TMACK1 TMACK0	Clock source selection 000: fpll-div 001: fpll-div/2 010: fpll-div/4 011: fpll-div/8 100: fpll-div/16 101: fpll-div/32 110: fs/2 111: fs/4

- Timer A Mode Register 2 (TMAMD2: 0x03F77)

bp	7	6	5	4	3	2	1	0
Flag	Reserved	PSCEN	-	-	-	-	-	-
At reset	0	0	-	-	-	-	-	-
Access	R/W	R/W	-	-	-	-	-	-

bp	Flag	Description
7	Reserved	Always set to “0”.
6	PSCEN	Prescaler operation control 0: Disable 1: Enable
5 to 0	-	-

When the TMAEN flag and PSCEN flag are set to “1”, the timer starts counting.

7.3 8-bit Simple Timer Count

7.3.1 8-bit Simple Timer Operation

8-bit simple timer contains one timer as an auxiliary function of 8-bit timers which are described in [Chapter 6 8-bit Timers].

- 8-bit Simple Timer Operation (Timer A)

The fundamental cycle of timer count is set based on the clock source selection and the setting value of TMAOC register, in advance. If TMABC reaches the setting value of TMAOC register, TMABC is cleared at the next count clock and counting is restarted from 0x00. This timer has no interrupt function.

The following table shows clock source that can be selected by timer.

Table:7.3.1 Clock Sources

Clock source	Time per Count	Timer A (8-bit)
fpll-div	50 ns	√
fpll-div/2	100 ns	√
fpll-div/4	200 ns	√
fpll-div/8	400 ns	√
fpll-div/16	0.8 µs	√
fpll-div/32	1.6 µs	√
fs/2	200 ns	√
fs/4	400 ns	√

fpll-div=20 MHz
fs=fpll-div/2=10 MHz

■ Count Timing of Timer Operation (Timer A)

Binary counter counts up with selected clock source as a count clock. The basic operation of the whole function of 8-bit timer is as follows:

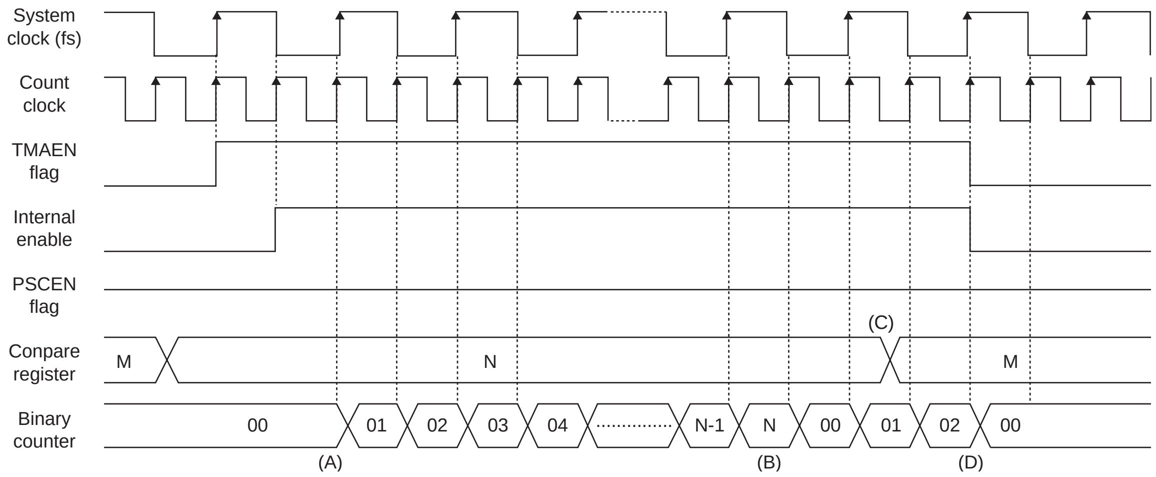


Figure:7.3.1 Count Timing of Timer Operation (Timer A)

- (A) When the TMAEN flag and the PSCEN flag start to operate ("1"), the internal enable will be turned on at the next count clock. Then binary counter begins counting up.
- (B) If binary counter reaches the value of compare register, binary counter is cleared to 0x00 at the next count clock and the counting restarts.
- (C) Even if compare register is rewritten while the TMAEN flag is enabled ("1"), binary counter is not changed.
- (D) When the TMAEN flag stops operating ("0"), the internal enable will be turned off. As a result, binary counter is cleared.

Switch the count clock after the timer operation is stopped, since the counting is not generated correctly during the timer operation.

Do not change the TMAEN flag of TMAMD1 register simultaneously with other bits to avoid operational errors.

When binary counter reaches the value in compare register, the value of the peripheral function count clock is inverted at the next count clock. So set compare register as:

Compare register setting = (Count till the compare match -1)

If compare register is set to smaller number than binary counter during the count operation, binary counter counts up until it overflows.

7.4 Serial Transfer Clock Output

7.4.1 Operation

8-bit simple timer can generate serial transfer clock using timer output signal.

Serial transfer clock operation by 8-bit timer (Timer A)

- Timer A: Serial interface 0, 1, 2, 4

■ Timing of Serial Transfer Clock (Timer A)

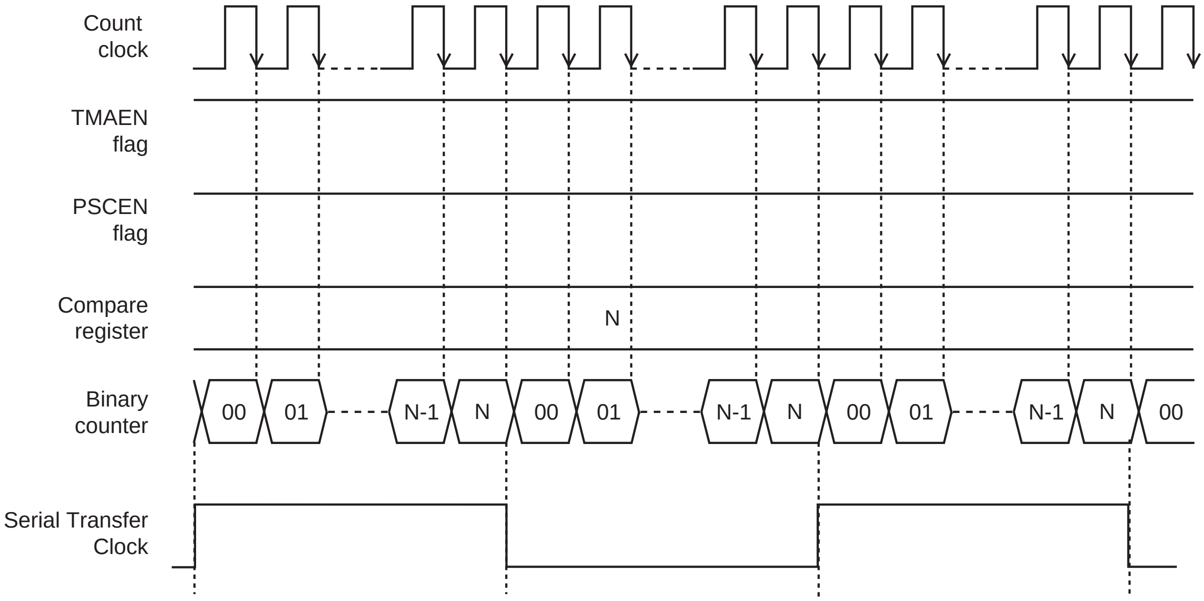


Figure:7.4.1 Timing of Serial Transfer Clock (Timer A)

- The timer frequency is 1/2 of the frequency set by compare register.
- For the baud rate calculation and the serial interface setup, refer to [Chapter 13 Serial Interface].

7.4.2 Setup Example

■ Serial Transfer Clock Setup Example (Timer A)

Transfer clock for full duplex UART (Serial 0) is generated by using Timer A.
The baud rate is selected to be 300 bps and the source clock of Timer A is selected to be fs/2 (at fs=2 MHz).

An example setup procedure, with a description of each step is shown below.

Setup Procedure	Description
(1) Stop the counter TMAMD1(0x03F75) bp3: TMAEN =0	(1) Set the TMAEN flag of TMAMD1 register to "0" to stop counter. When the counter is stopped, TMABC is cleared to 0x00.
(2) Select the count clock source TMAMD1(0x03F75) bp2 to 0: TMACK2 to 0 =110	(2) Select fs/2 as clock source by the TMACK2 to 0 flags of TMAMD1 register.
(3) Set the baud rate TMAOC (0x03F73) =0xCF	(3) Set a value to set the baud rate for TMAOC register to be 300 bps.
(4) Enable the prescaler count TMAMD2(0x03F77) bp6: PSCEN =1	(4) Set the PSCEN flag of TMAMD2 register to "1" to start prescaler.
(5) Start the timer operation TMAMD1(0x03F75) bp3: TMAEN =1	(5) Set the TMAEN flag of TMAMD1 register to "1" to start Timer A.

- TMABC counts up from 0x00. The output from Timer A will be a clock source of Serial Interface 0 at transmission and reception.
- For setting value of compare register and setup for serial interface operation, refer to [Chapter 13 Serial Interface].

Chapter 8　16-bit Timers

8.1 Overview

This LSI consists of one general-purpose 16-bit timer (Timer 7 and Timer 8). The 16-bit timer equipped with double-buffered compare register.

Timer n (high precision 16-bit timer) contains 2 sets of double-buffered compare register and 2 sets of independent interrupt functions such as Timer n interrupt and Timer n compare register 2 match interrupt.

Pins can be switched to TMnIOA/TMnIOB.

Table:8.1.1 16-bit Timer Pin Functions

Table remarks √: With function -: Without function

Functions	Pin Name	MN101EFA8	MN101EFA3	MN101EFA7	MN101EFA2
TM7IOA	PA5	√	√	√	√
TM7IOB	P02	√	√	√	√
TM8IOA	PA6	√	√	√	√
TM8IOB	P03	√	√	√	√

In this manual, if there is not much difference in the function between Pin A and B, "A" and "B” of the pin names are omitted.

Be sure to halt the 16-bit timer function before changing the frequency of fpll-div by bp7 to bp4 of OSCCNT register.

8.1.1 Functions

Table:8.1.2 shows the functions of each timer.

Table:8.1.2 16-bit Timer Functions

	Timer 7 (High-precision 16-bit timer)	Timer 8 (High-precision 16-bit timer)
Interrupt source	TM7IRQ	TM8IRQ
	TM7OC2IRQ	TM8OC2IRQ
Timer operation	√	√
Event count	TM7IOA input	TM8IOA input
	TM7IOB input	TM8IOB input
Timer pulse output	TM7IOA output	TM8IOA output
	TM7IOB output	TM8IOB output
PWM output (duty is changeable)	TM7IOA output	TM8IOA output
	TM7IOB output	TM8IOB output
High precision PWM output (duty/cycle are changeable)	TM7IOA output	TM8IOA output
	TM7IOB output	TM8IOB output
Capture function	√	√
Pulse width measurement	√	√
Clock source	fpll-div	fpll-div
	fpll-div/2	fpll-div/2
	fpll-div/4	fpll-div/4
	fpll-div/16	fpll-div/16
	fs	fs
	fs/2	fs/2
	fs/4	fs/4
	fs/16	fs/16
	TM7IO input	TM7IO input
	TM7IO input/2	TM7IO input/2
	TM7IO input/4	TM7IO input/4
	TM7IO input/16	TM7IO input/16
	Synchronous TM7IO input	Synchronous TM7IO input
	Synchronous TM7IO input/2	Synchronous TM7IO input/2
	Synchronous TM7IO input/4	Synchronous TM7IO input/4
	Synchronous TM7IO input/16	Synchronous TM7IO input/16
	Timer A output	Timer A output
	Timer A output/2	Timer A output/2
	Timer A output/4	Timer A output/4
	Timer A output/16	Timer A output/16

fpll-div: Machine clock (High speed oscillation for peripheral functions)
fs: System clock

8.1.2 Block Diagram

■ Timer 7 Block Diagram

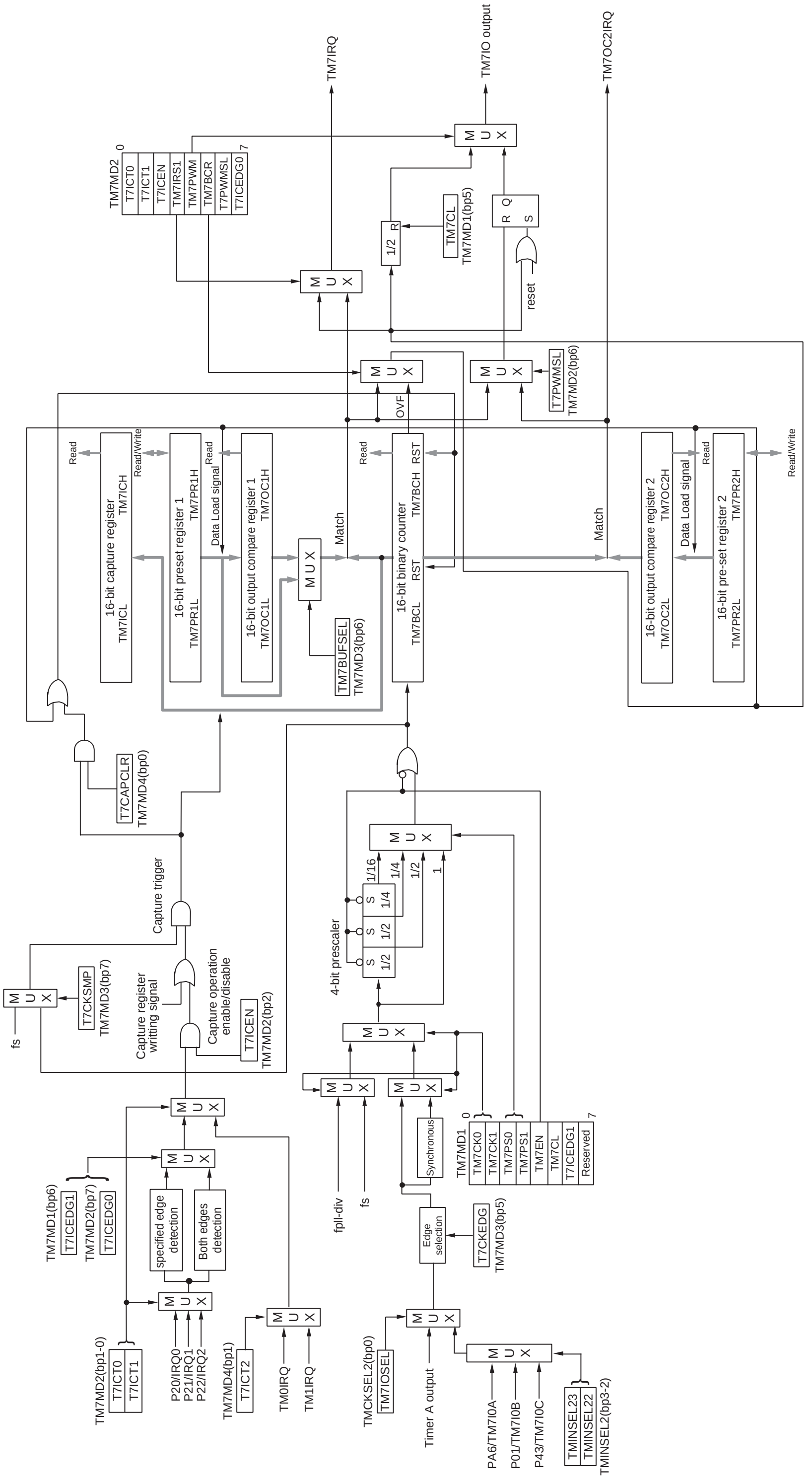


Figure:8.1.1 Timer 7 Block Diagram

■ Timer 8 Block Diagram

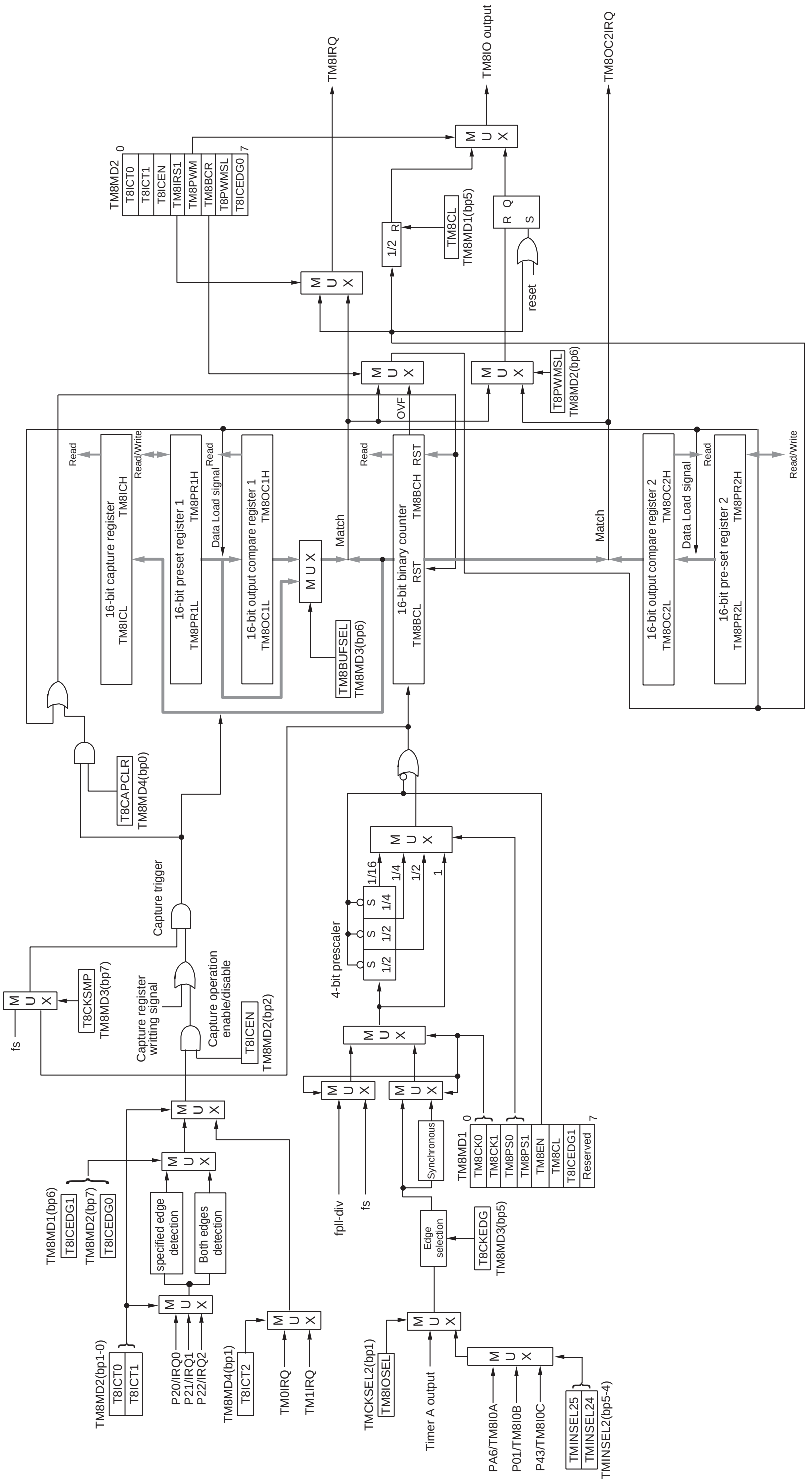


Figure:8.1.2 Timer 8 Block Diagram

8.2 Control Registers

Timer 7 and 8 consists of binary counter (TM7BC, TM8BC), compare register 1 (TM7OC1, TM8OC1) and its double-buffered preset register 1 (TM7PR1, TM8PR1), compare register 2 (TM7OC2, TM8OC2) and its double-buffered preset register 2 (TM7PR2, TM8PR2), and capture register (TM7IC, TM8IC).
Timer 7 and 8 is controlled by mode register 1 (TM7MD1, TM8MD1), mode register 2 (TM7MD2, TM8MD2), mode register 3 (TM7MD3, TM8MD3) and mode register 4 (TM7MD4, TM8MD4).

8.2.1 Registers

Table:8.2.1 shows the registers that control timer 7 and 8.

Table:8.2.1 16-bit Timer Control Registers

Table remarks √: With function -: Without function

	Register	Address	R/W	Function	Page	MN101 EFA8	MN101 EFA3	MN101 EFA7	MN101 EFA2
Timer 7	TM7BCL	0x03F80	R	Timer 7 binary counter (lower 8 bits)	VIII-11	√	√	√	√
	TM7BCH	0x03F81	R	Timer 7 binary counter (upper 8 bits)	VIII-11	√	√	√	√
	TM7OC1L	0x03F82	R	Timer 7 compare register 1 (lower 8 bits)	VIII-9	√	√	√	√
	TM7OC1H	0x03F83	R	Timer 7 compare register 1 (upper 8 bits)	VIII-9	√	√	√	√
	TM7PR1L	0x03F84	R/W	Timer 7 preset register 1 (lower 8 bits)	VIII-10	√	√	√	√
	TM7PR1H	0x03F85	R/W	Timer 7 preset register 1 (upper 8 bits)	VIII-10	√	√	√	√
	TM7ICL	0x03F86	R	Timer 7 input capture register (lower 8 bits)	VIII-11	√	√	√	√
	TM7ICH	0x03F87	R	Timer 7 input capture register (upper 8 bits)	VIII-11	√	√	√	√
	TM7MD1	0x03F88	R/W	Timer 7 mode register 1	VIII-12	√	√	√	√
	TM7MD2	0x03F89	R/W	Timer 7 mode register 2	VIII-13	√	√	√	√
	TM7OC2L	0x03F8A	R	Timer 7 compare register 2 (lower 8 bits)	VIII-9	√	√	√	√
	TM7OC2H	0x03F8B	R	Timer 7 compare register 2 (upper 8 bits)	VIII-9	√	√	√	√
	TM7PR2L	0x03F8C	R/W	Timer 7 preset register 2 (lower 8 bits)	VIII-10	√	√	√	√
	TM7PR2H	0x03F8D	R/W	Timer 7 preset register 2 (upper 8 bits)	VIII-10	√	√	√	√
	TM7MD3	0x03F9E	R/W	Timer 7 mode register 3	VIII-14	√	√	√	√
	TM7MD4	0x03F7E	R/W	Timer 7 mode register 4	VIII-14	√	√	√	√
	TM7ICR	0x03FF5	R/W	Timer 7 interrupt	IV-24	√	√	√	√
	TM7OC2ICR	0x03FF6	R/W	Timer 7 compare 2-match interrupt	IV-25	√	√	√	√
	TMCKSEL2	0x03FB1	R/W	Timer clock selection register 2	VIII-15	√	√	√	√
	TMINSEL2	0x03FB3	R/W	Timer input selection register 2	VIII-16	√	√	√	√

R/W: Readable/Writable
R: Read only

Table remarks √: With function -: Without function

	Register	Address	R/W	Function	Page	MN101 EFA8	MN101 EFA3	MN101 EFA7	MN101 EFA2
Timer 8	TM8BCL	0x03F90	R	Timer 8 binary counter (lower 8 bits)	VIII-11	√	√	√	√
	TM8BCH	0x03F91	R	Timer 8 binary counter (upper 8 bits)	VIII-11	√	√	√	√
	TM8OC1L	0x03F92	R	Timer 8 compare register 1 (lower 8 bits)	VIII-9	√	√	√	√
	TM8OC1H	0x03F93	R	Timer 8 compare register 1 (upper 8 bits)	VIII-9	√	√	√	√
	TM8PR1L	0x03F94	R/W	Timer 8 preset register 1 (lower 8 bits)	VIII-10	√	√	√	√
	TM8PR1H	0x03F95	R/W	Timer 8 preset register 1 (upper 8 bits)	VIII-10	√	√	√	√
	TM8ICL	0x03F96	R	Timer 8 input capture register (lower 8 bits)	VIII-11	√	√	√	√
	TM8ICH	0x03F97	R	Timer 8 input capture register (upper 8 bits)	VIII-11	√	√	√	√
	TM8MD1	0x03F98	R/W	Timer 8 mode register 1	VIII-12	√	√	√	√
	TM8MD2	0x03F99	R/W	Timer 8 mode register 2	VIII-13	√	√	√	√
	TM8OC2L	0x03F9A	R	Timer 8 compare register 2 (lower 8 bits)	VIII-9	√	√	√	√
	TM8OC2H	0x03F9B	R	Timer 8 compare register 2 (upper 8 bits)	VIII-9	√	√	√	√
	TM8PR2L	0x03F9C	R/W	Timer 8 preset register 2 (lower 8 bits)	VIII-10	√	√	√	√
	TM8PR2H	0x03F9D	R/W	Timer 8 preset register 2 (upper 8 bits)	VIII-10	√	√	√	√
	TM8MD3	0x03F9F	R/W	Timer 8 mode register 3	VIII-14	√	√	√	√
	TM8MD4	0x03F7F	R/W	Timer 8 mode register 4	VIII-14	√	√	√	√
	TM8ICR	0x03FF7	R/W	Timer 8 interrupt	IV-24	√	√	√	√
	TM8OC2ICR	0x03FF8	R/W	Timer 8 compare 2-match interrupt	IV-25	√	√	√	√
	TMCKSEL2	0x03FB1	R/W	Timer clock selection register 2	VIII-15	√	√	√	√
	TMINSEL2	0x03FB3	R/W	Timer input selection register 2	VIII-16	√	√	√	√

R/W: Readable/Writable
R: Read only

8.2.2 Programmable Timer Registers

Timer 7 and 8 have a set of 16-bit programmable timer registers, which contains a compare register, a preset register, a binary counter and a capture register. Each register has 2 sets of 8-bit register. They can be operated by 16-bit access.

A compare register is a 16-bit register which stores the values which will use to compare with binary counter.

- Timer n Compare Register 1 Lower 8 bits (TM7OC1L: 0x03F82, TM8OC1L: 0x03F92)

bp	7	6	5	4	3	2	1	0
Flag	TMnOC1L7	TMnOC1L6	TMnOC1L5	TMnOC1L4	TMnOC1L3	TMnOC1L2	TMnOC1L1	TMnOC1L0
At reset	X	X	X	X	X	X	X	X
Access	R	R	R	R	R	R	R	R

- Timer n Compare Register 1 Upper 8 bits (TM7OC1H: 0x03F83, TM8OC1H: 0x03F93)

bp	7	6	5	4	3	2	1	0
Flag	TMnOC1H7	TMnOC1H6	TMnOC1H5	TMnOC1H4	TMnOC1H3	TMnOC1H2	TMnOC1H1	TMnOC1H0
At reset	X	X	X	X	X	X	X	X
Access	R	R	R	R	R	R	R	R

- Timer n Compare Register 2 Lower 8 bits (TM7OC2L: 0x03F8A, TM8OC2L: 0x03F9A)

bp	7	6	5	4	3	2	1	0
Flag	TMnOC2L7	TMnOC2L6	TMnOC2L5	TMnOC2L4	TMnOC2L3	TMnOC2L2	TMnOC2L1	TMnOC2L0
At reset	X	X	X	X	X	X	X	X
Access	R	R	R	R	R	R	R	R

- Timer n Compare Register 2 Upper 8 bits (TM7OC2H: 0x03F8B, TM8OC2H: 0x03F9B)

bp	7	6	5	4	3	2	1	0
Flag	TMnOC2H7	TMnOC2H6	TMnOC2H5	TMnOC2H4	TMnOC2H3	TMnOC2H2	TMnOC2H1	TMnOC2H0
At reset	X	X	X	X	X	X	X	X
Access	R	R	R	R	R	R	R	R

Timer 7 and 8 preset registers 1, 2 are buffer registers of the compare registers 1 and 2 of Timer 7 and 8. If the set value is written to Timer 7 and 8 preset registers 1 and 2 when the counting is stopped, the same set value is loaded to Timer 7 and 8 compare register. If the set value is written to Timer 7 and 8 preset registers 1 and 2 during counting, the set value of Timer 7 and 8 preset registers 1 and 2 is loaded to Timer 7 and 8 compare registers 1 and 2 at the timing that Timer 7 and 8 binary counter are cleared.

- Timer n Preset Register 1 Lower 8 bits (TM7PR1L: 0x03F84, TM8PR1L: 0x03F94)

bp	7	6	5	4	3	2	1	0
Flag	TMnPR1L7	TMnPR1L6	TMnPR1L5	TMnPR1L4	TMnPR1L3	TMnPR1L2	TMnPR1L1	TMnPR1L0
At reset	X	X	X	X	X	X	X	X
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- Timer n Preset Register 1 Upper 8 bits (TM7PR1H: 0x03F85, TM8PR1H: 0x03F95)

bp	7	6	5	4	3	2	1	0
Flag	TMnPR1H7	TMnPR1H6	TMnPR1H5	TMnPR1H4	TMnPR1H3	TMnPR1H2	TMnPR1H1	TMnPR1H0
At reset	X	X	X	X	X	X	X	X
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- Timer n Preset Register 2 Lower 8 bits (TM7PR2L: 0x03F8C, TM8PR2L: 0x03F9C)

bp	7	6	5	4	3	2	1	0
Flag	TMnPR2L7	TMnPR2L6	TMnPR2L5	TMnPR2L4	TMnPR2L3	TMnPR2L2	TMnPR2L1	TMnPR2L0
At reset	X	X	X	X	X	X	X	X
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- Timer n Preset Register 2 Upper 8 bits (TM7PR2H: 0x03F8D, TM8PR2H: 0x03F9D)

bp	7	6	5	4	3	2	1	0
Flag	TMnPR2H7	TMnPR2H6	TMnPR2H5	TMnPR2H4	TMnPR2H3	TMnPR2H2	TMnPR2H1	TMnPR2H0
At reset	X	X	X	X	X	X	X	X
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Under the following conditions, update the value of Timer 7 and 8 preset register 1 or 2 after setting the TMnCK1 to 0 flags of Timer 7 and 8 mode register 1 to “01” (clock source: fs). And then switch the clock source to be used and operate the timer.

<Conditions>

The TMnBUFSEL flag of TMnMD3 register = 0 (double buffer),
the TMnEN flag of TMnMD1 register = 0 (count stop),
the TMnCK1 to 0 flags of TMnMD1 register = 00 (fpll-div) and
[system clock (fs) > high-speed clock for peripheral function (pll-div) × 2]

Binary counter is a 16-bit up counter. If the set value is written to preset register when the counting is stopped, binary counter is cleared to 0x0000. Also, by setting TnCAPCLR, binary counter is cleared to 0x0000 at capture.

- Timer n Binary Counter Lower 8 bits (TM7BCL: 0x03F80, TM8BCL: 0x03F90)

bp	7	6	5	4	3	2	1	0
Flag	TMnBCL7	TMnBCL6	TMnBCL5	TMnBCL4	TMnBCL3	TMnBCL2	TMnBCL1	TMnBCL0
At reset	X	X	X	X	X	X	X	X
Access	R	R	R	R	R	R	R	R

- Timer n Binary Counter Upper 8 bits (TM7BCH: 0x03F81, TM8BCH: 0x03F91)

bp	7	6	5	4	3	2	1	0
Flag	TMnBCH7	TMnBCH6	TMnBCH5	TMnBCH4	TMnBCH3	TMnBCH2	TMnBCH1	TMnBCH0
At reset	X	X	X	X	X	X	X	X
Access	R	R	R	R	R	R	R	R

Input capture register is a register that holds the value loaded from binary counter by the capture trigger.
The capture trigger is generated by external interrupt, Timer 0 interrupt, Timer 1 interrupt and when an arbitrary value is written to input capture register. (Directly writing to the register by program is disabled.)

- Timer n Input Capture Register Lower 8 bits (TM7ICL: 0x03F86, TM8ICL: 0x03F96)

bp	7	6	5	4	3	2	1	0
Flag	TMnICL7	TMnICL6	TMnICL5	TMnICL4	TMnICL3	TMnICL2	TMnICL1	TMnICL0
At reset	X	X	X	X	X	X	X	X
Access	R	R	R	R	R	R	R	R

- Timer n Input Capture Register Upper 8 bits (TM7ICH: 0x03F87, TM8ICH: 0x03F97)

bp	7	6	5	4	3	2	1	0
Flag	TMnICH7	TMnICH6	TMnICH5	TMnICH4	TMnICH3	TMnICH2	TMnICH1	TMnICH0
At reset	X	X	X	X	X	X	X	X
Access	R	R	R	R	R	R	R	R

8.2.3 Timer Mode Registers

These are readable/writable registers that control Timer 7 and 8.

- Timer n Mode Register 1 (TM7MD1: 0x03F88, TM8MD1: 0x03F98)

bp	7	6	5	4	3	2	1	0
Flag	Reserved	TnICEDG1	TMnCL	TMnEN	TMnPS1	TMnPS0	TMnCK1	TMnCK0
At reset	0	0	1	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7	Reserved	Always set to “0”.
6	TnICEDG1	Capture trigger edge selection 0: Falling edge 1: Rising edge
5	TMnCL	Timer output reset signal 0: Operate timer output 1: Disable timer output (reset)
4	TMnEN	Timer count control 0: Halt the count 1: Operate the count
3-2	TMnPS1-0	Count clock selection 00: 1/1 of clock 01: 1/2 of clock 10: 1/4 of clock 11: 1/16 of clock
1-0	TMnCK1-0	Clock source selection 00: fpll-div 01: fs 10: TMIO input 11: Synchronous TMIO input

When using TMnIO input, set each mode register and preset register after selecting fs as count clock. Then, select TMnIO input to start the timer. Do not write any data to preset register while the timer is operating. 16-bit timer can return from STOP mode only with TMnIO input.

■ Timer n Mode Register 2 (TM7MD2: 0x03F89, TM8MD2: 0x03F99)

bp	7	6	5	4	3	2	1	0
Flag	TnICEDG0	TnPWMSL	TMnBCR	TMnPWM	TMnIRS1	TnICEN	TnICT1	TnICT0
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7	TnICEDG0	Capture trigger edge selection 0: Select the both edges 1: Select the specified edge
6	TnPWMSL	PWM mode selection 0: Set duty by OC1 1: Set duty by OC2
5	TMnBCR	Timer n count clear factor selection 0: Full count OVF 1: Match of BC and OC1
4	TMnPWM	Timer output waveform selection 0: Output timer 1: Output PWM
3	TMnIRS1	Timer n interrupt factor selection 0: Counter clear 1: Match of BC and OC1
2	TnICEN	Input capture operation enable select flag 0: Disable capture operation 1: Enable capture operation
1-0	TnICT1-0	Capture trigger selection 00: External interrupt 0 input signal 01: External interrupt 1 input signal 10: External interrupt 2 input signal 11: Timer interrupt

When TnICT1 to 0 flags are set to “Timer interrupt”, the timer selected by TnICT2 flag of TMnMD4 register is set as a capture trigger.

- Timer n Mode Register 3 (TM7MD3: 0x03F9E, TM8MD3: 0x03F9F)

bp	7	6	5	4	3	2	1	0
Flag	TMn CKSMP	TMn BUFSEL	TMn CKEDG	-	-	-	-	-
At reset	0	0	0	-	-	-	-	-
Access	R/W	R/W	R/W	-	-	-	-	-

bp	Flag	Description
7	TMnCKSMP	Capture sampling selection 0: Count clock 1: fs
6	TMnBUFSEL	Buffer selection 0: Double buffer 1: Single buffer
5	TMnCKEDG	TMnIO count edge selection 0: Falling edge 1: Both edges
4-0	-	-

- Timer n Mode Register 4 (TM7MD4: 0x03F7E, TM8MD4: 0x03F7F)

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	-	-	Tn ICT2	Tn CAPCLR
At reset	-	-	-	-	-	-	0	0
Access	-	-	-	-	-	-	R/W	R/W

bp	Flag	Description
7-2	-	-
1	TnICT2	Capture trigger selection 0: Timer 0 interrupt 1: Timer 1 interrupt
0	TnCAPCLR	TMnBC clearing at capture 0: Not cleared 1: Cleared

TnCAPCLR flag is valid when timer is operating.
Note that binary counter is not cleared when data is being captured during timer stop.

- Timer Clock Selection Register 2 (TMCKSEL2: 0x03FB1)

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	-	TM9IOSEL	TM8IOSEL	TM7IOSEL
At reset	-	-	-	-	-	0	0	0
Access	-	-	-	-	-	R/W	R/W	R/W

bp	Flag	Description
7-3	-	-
2	TM9IOSEL	Timer 9 input selection 0: External input 1: Timer A
1	TM8IOSEL	Timer 8 input selection 0: External input 1: Timer A
0	TM7IOSEL	Timer 7 input selection 0: External input 1: Timer A

■ Timer Input Selection Register 2 (TMINSEL2: 0x03FB3)

bp	7	6	5	4	3	2	1	0
Flag	TMINSEL 27	TMINSEL 26	TMINSEL 25	TMINSEL 24	TMINSEL 23	TMINSEL 22	-	-
At reset	0	0	0	0	0	0	-	-
Access	R/W	R/W	R/W	R/W	R/W	R/W	-	-

bp	Flag	Description
7-6	TMINSEL27-26	Timer 9 input selection (at port selection) 00: TM9IOA 01: TM9IOB 1x: ”1” is input to timer
5-4	TMINSEL25-24	Timer 8 input selection (at port selection) 00: TM8IOA 01: TM8IOB 1x: ”1” is input to timer
3-2	TMINSEL23-22	Timer 7 input selection (at port selection) 00: TM7IOA 01: TM7IOB 1x: ”1” is input to timer
1-0	-	-

8.3 Operation

8.3.1 Operation

This timer has a function to generate interrupts constantly in a certain period of time.

- 16-bit Timer Operation (Timer 7 and 8)

The generation cycle of a timer interrupt is set by the clock source selection and the set value of TMnOC1 register, in advance. When TMnBC reaches the set value of TMnOC1 register, an interrupt is generated at the next count clock. There are 2 factors to be selected to clear binary counter; TMnOC1 compare match and the full count overflow. After binary counter is cleared, the counting up is restarted from 0x0000.

Table:8.3.1 16-bit Timer Interrupt Factor and Binary Counter Clear Factor (Timer 7 and 8)

TMnMD2 register		Interrupt factor	Binary counter clear factor
TMnIRS1	TMnBCR		
1	1	TMnOC1 compare match	TMnOC1 compare match
0	1	TMnOC1 compare match	TMnOC1 compare match
1	0	TMnOC1 compare match	Full count overflow
0	0	Full count overflow	Full count overflow

Timer 7 and 8 can generate another independent interrupt (Timer 7 and 8 compare register 2 match interrupt) by the set value of TMnOC2 register. When this interrupt is generated, binary counter is cleared as indicated above.

Compare register is double-buffered type. When the value of preset registers is changed during counting operation, the value changed is stored to compare register when binary counter is cleared. This function can change compare register value constantly, without disturbing the cycle during timer operation (Reload function).

When CPU reads 16-bit binary counter (TMnBC), the read data is handled in 8-bit units even if it is a 16-bit MOVW instruction. As a result, it will read the data incorrectly if a carry from the lower 8 bits to the upper 8-bit occurs during counting operation.
To read the correct value of 16-bit counting (TMnBC), use the write program function to write to input capture register (TMnIC). By writing to TMnIC, counting data of TMnBC are stored to TMnIC so that the values can be read out correctly during operation.
[Chapter 8 8.3.1 Operation]

To count properly, do not switch the count clock during the timer operation.
To switch the count clock, stop the timer operation.

Set timer mode register when the TMnEN flag of TMnMD1 register is set to "0" to stop counting.

Use the MOVW instruction of 16-bit access to write the data to preset register (TMnPR1 and TMnPR2) when 16-bit timer is stopped.
If lower 8-bit is written after upper 8-bit is written to preset register by using the MOV instruction of 8-bit access, the rewritten data is loaded to upper 8-bit and data which is not written is loaded to lower 8-bit of compare register (TMnOC1 and TMnOC2).

Table:8.3.2 shows the clock source that can be selected.

Table:8.3.2 Clock Source at Timer Operation (Timer 7 and Timer 8)

Clock source	Time per count
fpll-div	100 ns
fpll-div/2	200 ns
fpll-div/4	400 ns
fpll-div/16	1.6 µs
fs	200 ns
fs/2	400 ns
fs/4	800 ns
fs/16	3.2 µs

fpll-div = 10 MHz (PLL is not used)
fs = fpll-div/2 = 5 MHz

■ Count Timing of Timer Operation (Timer 7 and 8)

The binary counter counts up with the selected clock source as the count clock.
The basic operation of whole 16-bit timer functions is as below.

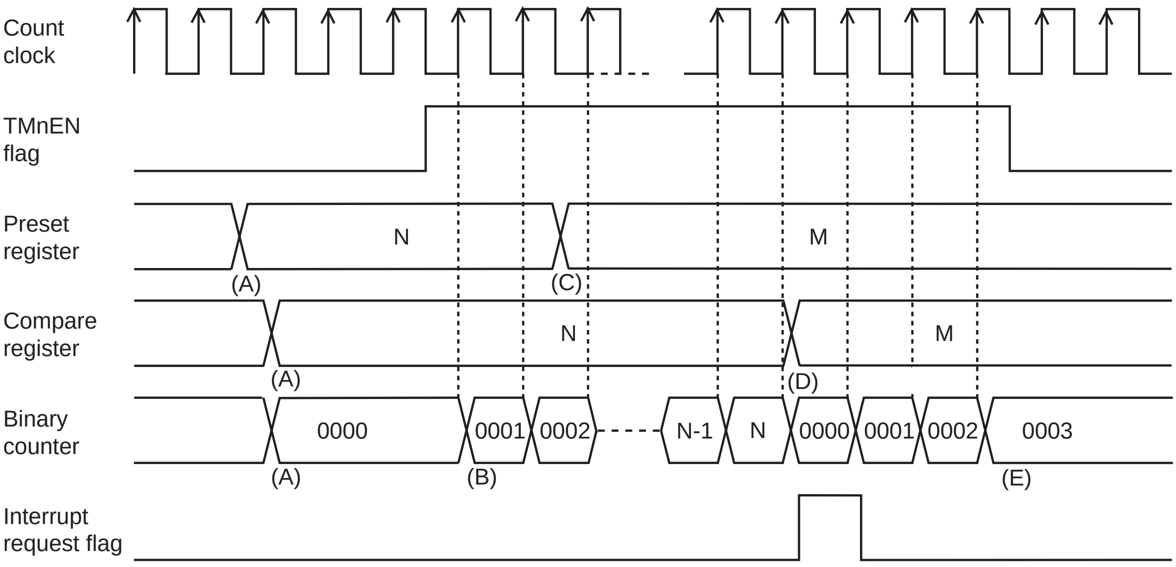


Figure:8.3.1 Count Timing of Timer Operation (Timer 7 and 8)

(A) When a data is written to preset register while the TMnEN flag is stopped ("0"), the same value is loaded during the writing cycle and binary counter is cleared to 0x0000.

(B) When TMnEN flag is "1", binary counter starts counting. The counting starts at the rising edge of the count clock.

(C) Even if preset register is rewritten when the TMnEN flag is "1", binary counter is not changed.

(D) When binary counter reaches value of compare register 1, the set value of preset register is loaded to compare register at the next count clock. And the interrupt request flag is set at the next count clock, and binary counter is cleared to 0x0000 to restart counting up.

(E) When the TMnEN flag is "0", binary counter is stopped.

When binary counter reaches the value of compare register, the interrupt request flag is set at the next count clock, and binary counter is cleared.
So, set compare register as:
(the set value of compare register) = (the counts till the interrupt generation - 1)
However, if “00” is specified for compare register, an interrupt timing is the same as if you set it to "01".

After Timer n interrupt request flag is generated, up to 3 system clock are needed until the next interrupt request flag is generated. During this period, an interrupt request flag is not generated even if a compare match occurs.

When Timer n compare register 2 match interrupt is generated and TMnOC1 compare match is selected as a binary counter clear source, the set value of compare register 2 should be smaller than the set value of compare register 1.

When timer interrupt request flag has been already set, be sure to clear timer interrupt request flag before starting the timer.

When binary counter is used as a free-counter that counts 0x0000 to 0xFFFF, set 0xFFFF to compare register or set the TMnBCR flag of TMnMD2 register to "0".

Do not change TMnEN flag of TMnMD1 register simultaneously with other bits to avoid any error in operation.

Set the count clock of 16-bit timer while timer interrupt is disabled.

When binary counter is read during the timer operation, the data is processed as 8-bit unit data in LSI. If there occurs a carry-out from lower 8-bit to upper 8-bit during reading, correct values cannot be read out. Stop timer to read correct values.

8.3.2 Setup Example

■ Timer Operation Setup Example

Timer function can be set by using Timer 7 that generates interrupts constantly. Interrupt is generated in every 1000 cycles (200 µs) by selecting fpll-div/2 (fpll-div=10 MHz at operation) as a clock source.
An example setup procedure, with a description of each step is shown below.

Setup Procedure	Description
(1) Stop the counter TM7MD1(0x03F88) bp4: TM7EN =0	(1) Set the TM7EN flag of TM7MD1 register to “0” to stop Timer 7 counting.
(2) Disable the interrupt TM7ICR(0x03FF5) bp1: TM7IE =0	(2) Set the TM7IE flag of TM7ICR register to “0” to disable the interrupt.
(3) Select the timer clear source TM7MD2(0x03F89) bp5: TM7BCR =1	(3) Set the TM7BCR flag of TM7MD2 register to “1” to select the compare match to binary counter clear source.
(4) Select the count clock source TM7MD1 (0x03F88) bp1 to 0: TM7CK1 to 0 =00 bp3 to 2: TM7PS1 to 0 =01	(4) Select fpll-div to the clock source by the TM7CK1 to 0 flags of TM7MD1 register. In addition, select 1/2 fpll-div to the count clock source by the TM7PS1 to 0 flags.
(5) Set the interrupt generation cycle TM7PR1(0x03F85, 0x03F84) =0x03E7	(5) Set the interrupt generation cycle to TM7PR1 register. The set value should be 1000-1=999 (0x03E7) because the cycle is 1000. At the time, the same value is loaded to TM7OC1 register, and TM7BC is initialized to 0x0000.
(6) Set the interrupt level TM7ICR(0x03FF5) bp7 to 6: TM7LV1 to 0 =10	(6) Set the interrupt level by the TM7LV1 to 0 flags of TM7ICR register. If the interrupt request flag has already been set, clear the request flag. [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup]
(7) Enable the interrupt TM7ICR (0x03FF5) bp1: TM7IE =1	(7) Set the TM7IC flag of TM7ICR register to “1” to enable the interrupt.
(8) Start the timer operation TM7MD1 (0x03F88) bp4: TM7EN =1	(8) Set the TM7EN flag of TM7MD1 register to “1” to operate Timer 7.

TM7BC counts up from 0x0000. When TM7BC reaches the set value of TM7OC1 register, Timer 7 interrupt request flag is set at the next count clock and TM7BC turns to 0x0000 and counts up again.

8.4 16-bit Event Count

8.4.1 Operation

Event counting operation is classified into two types based on the clock source selected: TMnIO input and synchronous TMnIO input. Each event counting can select 1/1, 1/2, 1/4 or 1/16 as a count clock source. Also, it is possible to select the count edge. (the falling edge and the both edge at the normal operation are selectable)

■ 16-bit Event Count Operation (Timer 7 and 8)

Binary counter (TMnBC) counts the external signal input to pin TMnIO. If binary counter reaches the set value of compare register (TMnOC), an interrupt can be generated at the next count clock.

Table:8.4.1 Event Count Input Clock

Table remarks √: With function -: Without function

	Event count input source	MN101EFA8	MN101EFA3	MN101EFA7	MN101EFA2
Timer 7	TM7IOA input	√	√	√	√
	TM7IOB input	√	√	√	√
	Synchronous TM7IO input	√	√	√	√
	Timer A output	√	√	√	√
Timer 8	TM8IOA input	√	√	√	√
	TM8IOB input	√	√	√	√
	Synchronous TM8IO input	√	√	√	√
	Timer A output	√	√	√	√

■ Count Timing of TMnIO Input (Timer 7 and 8)

When TMnIO input is selected, TMnIO input signal is input to Timer n count clock.
Binary counter counts up at the falling edge of the TMnIO input signal or TMnIO input signal that passed the divider.

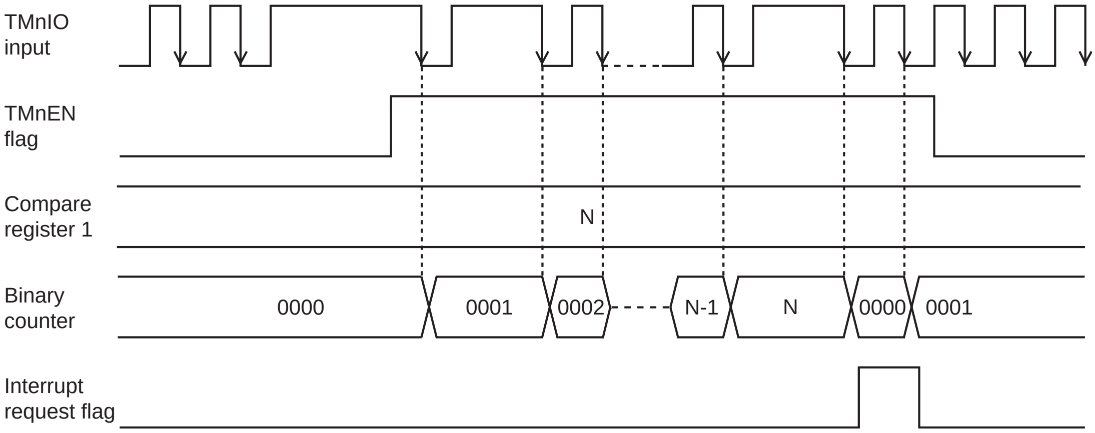


Figure:8.4.1 Count Timing TMnIO Input (Timer 7 and 8)

When the event input (TMnIO input) is selected as a count clock source, do not read out values of binary counter. To read out values of binary counter, use an event counter with synchronous TMnIO input which is indicated below.

When using TMnIO input, be sure to set each mode register and preset register after selecting fs as a count clock source. Then, select TMnIO input to start a timer. Do not write any data to preset register during operation. Only TMnIO input can recover from STOP mode in 16-bit timer.

When using the event input (TMnIO input), clear binary counter before starting the timer operation. Also, when setting 0x0000 to compare register, use the event count by synchronous TMnIO input which is shown below.

When the event input (TMnIO input) is selected as a count clock source, binary counter may reach an unexpected value by stopping timer operation. When the event input (TMnIO input) is selected as a count clock source, do not read binary counter.

■ Count Timing of Synchronous TMnIO Input (Timer 7 and 8)

If the synchronous TMnIO input is selected, the synchronizing circuit output signal is input to Timer n count clock. The synchronizing circuit output signal is changed at the rising edge of the system clock after the TMnIO input signal is changed. Binary counter counts up at the falling edge of the synchronizing circuit output signal or the synchronizing circuit output signal that passed through the division circuit.

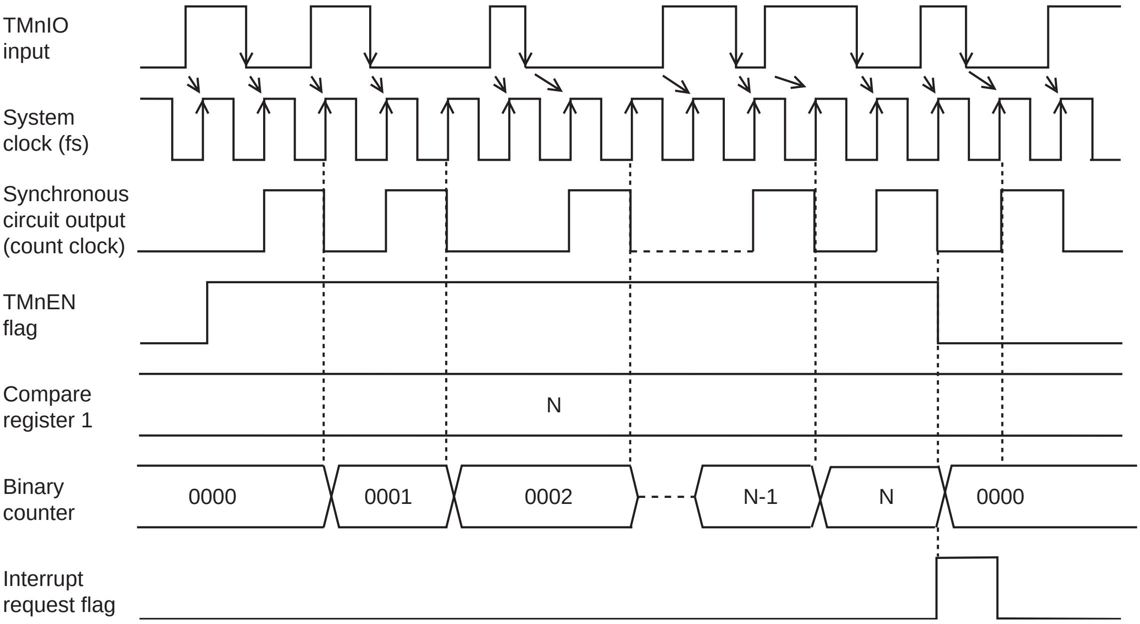


Figure:8.4.2 Count Timing of Synchronous TMnIO Input (Timer 7 and 8)

As Timer n counts up binary counter using signals synchronized to system clock, correct values are always read out when Timer n binary counter is read out.

Input signal from TMnIO should be set with the cycle more than twice the system clock (fs). When other signals with a cycle shorter than this are input, the counting may not be performed correctly.

■ Count Timing of TMnIO Input (Both edges selected)

When TMnIO input is selected, TMnIO input signal is input to Timer n count clock. Binary counter counts up at the rising or the falling edge of TMnIO input signal that passed the divider or TMnIO input signal.

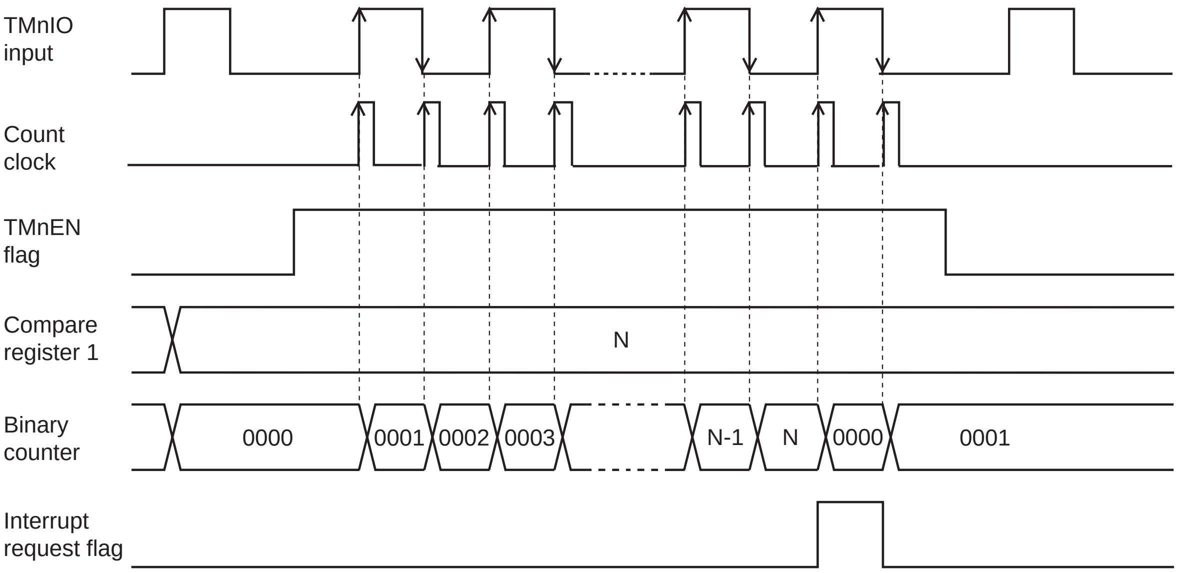


Figure:8.4.3 Count Timing of TMnIO Input (Timer 7 and 8)

When both edges are selected, they are counted only at the normal operation (high-speed oscillation). The counting can not be operated when synchronous TMnIO is set as input. Input from TMnIO should be done the waveform which has more than 2 times cycle than fpll-div. If less than the above waveforms are input, it may not be counted correctly.

8.4.2 Setup Example

■ Event Count Setup Example

When the falling edge of TM7IO input pin signal is detected 5 times using Timer 7, an interrupt is generated. An example setup procedure, with a description of each step is shown below.

Setup Procedure	Description
(1) Stop the counter TM7MD1 (0x03F88) bp4: TM7EN =0	(1) Set the TM7EN flag of TM7MD1 register to “0” to stop Timer 7 counting.
(2) Disable the interrupt TM7ICR(0x03FF5) bp1: TM7IE =0	(2) Set the TM7IE flag of TM7ICR register to “0 “to disable the interrupt.
(3) Select the timer clock TMCKSEL2(0x03FB1) bp0: TM7IOSEL =0	(3) Select the external input by setting the TM7IOSEL flag of TMCKSEL2 register to “0”.
(4) Select the timer input TMINSEL2(0x03FB3) bp3 to 2: TMINSEL23 to 22 =00	(4) Set PA6 to timer input by setting the TMINSEL23 to 22 flags of TMINSEL2 register to “00”.
(5) Set the special function pin to input PADIR (0x03E9A) bp5: PADIR5 =0	(5) Set the PADIR5 flag of PADIR register to “0” to set PA5 pin to the input mode. Add pull-up/pull-down resistor, according to need. [Chapter 5 I/O Port]
(6) Select the count clock source TM7MD1(0x03F88) bp1 to 0: TM7CK1 to 0 =01 bp3 to 2: TM7PS1 to 0 =00	(6) Select fs to the clock source by the TM7CK1 to 0 flags of TM7MD1 register. In addition, select 1/1 to the count clock source by the TM7PS1 to 0 flags.
(7) Set the interrupt generation cycle TM7PR1(0x03F85, 0x03F84) =0x0004	(7) Set the interrupt generation cycle to TM7PR1 register. As counting is executed 5 times, the set value should be 0x0004. At that time, the same value is loaded to TM7OC1 registerTM7OC1, and TM7BC is initialized to 0x0000.
(8) Select the timer clear source TM7MD2 (0x03F89) bp5: TM7BCR =1	(8) Set the TM7BCR flag of TM7MD2 register to “1” to select the compare match as a binary counter clear source.
(9) Select the count clock source TM7MD1 (0x03F88) bp1 to 0: TM7CK1 to 0 =10 bp3 to 2: TM7PS1 to 0 =00	(9) Select TM7IO to the clock source by the TM7CK1 to 0 flags of TM7MD1 register. In addition, select 1/1 to the count clock source by the TM7PS1 to 0 flags.
(10) Set the interrupt level TM7ICR (0x03FF5) bp7 to 6: TM7LV1 to 0 =10	(10) Set the interrupt level by the TM7LV1 to 0 flags of TM7ICR register. If the interrupt request flag has already been set, clear the request flag. [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup]
(11) Enable the interrupt TM7ICR (0x03FF5) bp1: TM7IE =1	(11) Set the TM7IE flag of TM7ICR register to “1” to enable the interrupt.

Setup Procedure	Description
(12) Start the event count TM7MD1 (0x03F88) bp4: TM7EN =1	(12) Set the TM7EN flag of TM7MD1 register to “1” to operate Timer 7.

Every time TM7BC reaches the falling edge of TM7IO input, it counts up from 0x0000.
When TM7BC reaches the set value of TM7OC1 register, Timer 7 interrupt request flag is set at the next count clock, and the value of TM7BC is 0x0000 to restart counting up.

Be sure to follow the procedures from (6) to (10) to avoid any error in operation.

8.5 16-bit Timer Pulse Output

8.5.1 Operation

Pin TMnIO can output pulse signals with arbitrary frequency.

- 16-bit Timer Pulse Output Operation (Timer 7 and 8)

Timer can output twice the cycle signal set to TMnOC1 register and output twice the cycle signal of the 16-bit full count. Table:8.5.1 shows timer pulse output pin.

Table:8.5.1 Timer Pulse Output Pin

Table remarks √: With function -: Without function

	Pulse output pin	MN101EFA8	MN101EFA3	MN101EFA7	MN101EFA2
Timer 7	TM7IOA output	√	√	√	√
	TM7IOB output	√	√	√	√
Timer 8	TM8IOA output	√	√	√	√
	TM8IOB output	√	√	√	√

Table:8.5.2 shows timer interrupt generation sources and the flags that control timer pulse output cycle.

Table:8.5.2 16-bit Timer Interrupt Generation Source and Timer Pulse Output Cycle (Timer 7 and 8)

TMnMD2 register		Interrupt source	Timer pulse output cycle
TMnIRS1 flag	TMnBCR flag		
1	1	TMnOC1 compare match	Set value of TMnOC1 × 2
0	1	TMnOC1 compare match	Set value of TMnOC1 × 2
1	0	TMnOC1 compare match	Full count of TMnBC × 2
0	0	Full count over flow	Full count of TMnBC × 2

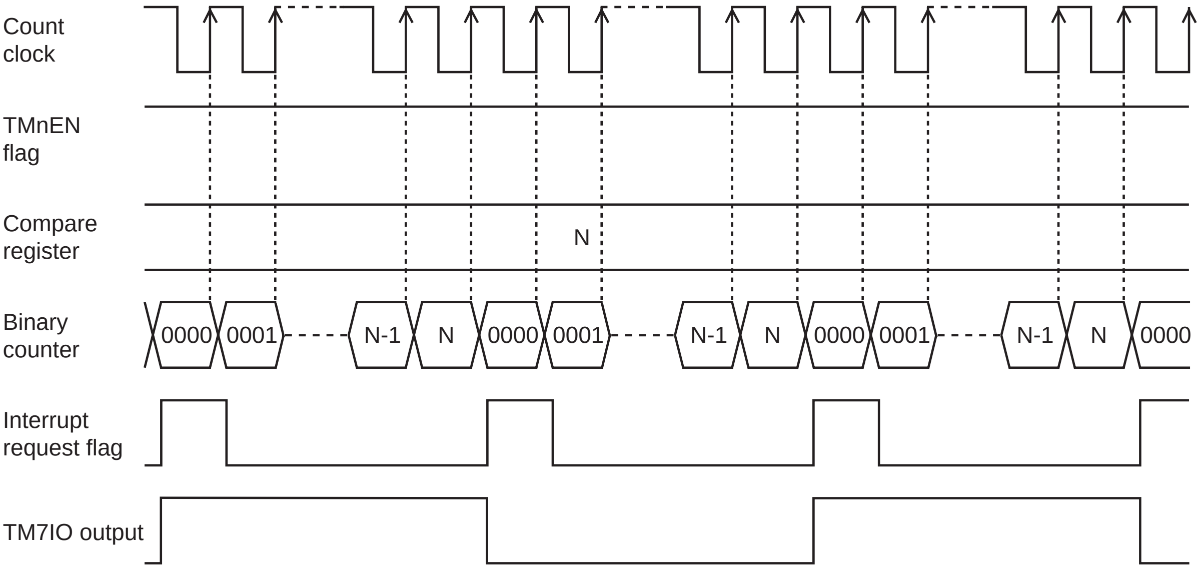


Figure:8.5.1 Count Timing of Timer Pulse Output (Timer 7 and 8)

TMnIO output pin twice the cycle signal set to TMnOC1 register. If binary counter reaches the compare value or full count overflow occurs, binary counter is cleared to 0x0000, and TMnIO output is inverted.

In the initial state after releasing reset, the timer pulse output is reset, and “Low” output is fixed. Therefore, release the reset of the timer pulse output by setting the TMnCL flag of TMnMD1 register to “0”.

Regardless of whether binary counter is stopped or operated, the timer output is "Low", when the TMnCL flag of TMnMD1 register is set to “1”.

Reset release of the timer pulse output should be done when the timer count is stopped.

To start timer pulse output with the divided clock source, set dividing ratio after releasing the reset of timer pulse output. Dividing ratio can be set by TMnPS1 to TMnPS0 flags of TMnMD1 register.

8.5.2 Setup Example

■ Timer Pulse Output Setup Example

TM7IO output pin outputs pulses of 25 kHz. Select fpll-div as the clock source and set 1/2 cycle (25 kHz) to Timer 7 compare register (at fpll-div=10 MHz).

An example setup procedure, with a description of each step is shown below.

Setup Procedure	Description
(1) Stop the counting TM7MD1 (0x03F88) bp4: TM7EN =0	(1) Set the TM7EN flag of TM7MD1 register to “0” to stop Timer 7 counting.
(2) Set the special function pin PAOMD (0x03EBA) bp5: PAOMD5 =1 PADIR (0x03E9A) bp5: PADIR5 =1	(2) Set the PAOMD5 flag of PAOMD register to “1” to set PA5 as the special function pin. Set the PADIR5 flag of PADIR register to “1” to set the output mode. [Chapter 5 I/O Port]
(3) Set the timer pulse TM7MD2 (0x03F89) bp4: TM7PWM =0	(3) Set the TM7PWM flag of TM7MD2 register to “0” to select the timer pulse output.
(4) Select the timer clear source TM7MD2 (0x03F89) bp5: TM7BCR =1	(4) Set the TM7BCR flag of TM7MD2 register to “1” to select the compare match as binary counter clear source.
(5) Release the reset of the timer pulse TM7MD1 (0x03F88) bp5: TM7CL =0	(5) Set the TM7CL flag of TM7MD1 register to “0” to enable the pulse output.
(6) Select the count clock source TM7MD1 (0x03F88) bp1 to 0: TM7CK1 to 0 =00 bp3 to 2: TM7PS1 to 0 =00	(6) Select fpll-div as the clock source by the TM7CK1 to 0 flags of TM7MD1 register. Also, select 1/1 dividing as the clock source by the TM7PS1 to 0 flags.
(7) Set the timer pulse output generation cycle TM7PR1(0x03F85, 0x03F84) =0x00C7	(7) Set 1/2 of the timer pulse output cycle to TM7PR1 register. To set 50 kHz by dividing 10 MHz, set as; 200-1=199 (0x00C7) At the same time, the same value is loaded to TM7OC1 register and TM7BC is initialized to 0x0000.
(8) Start the timer operation TM7MD1 (0x03F88) bp4: TM7EN =1	(8) Set the TM7EN flag of TM7MD1 register to “1” to operate Timer 7.

TM7BC counts up from 0x0000. If TM7BC reaches the set value of TM7OC1 register, and TM7BC is cleared to 0x0000, the signal of TM7IO output is inverted and TM7BC counts up from 0x0000 again.

8.6 16-bit Standard PWM Output (Only duty can be changed consecutively)

TM7IO pin outputs the standard PWM waveform, which is generated for either case when the values of both binary counter and compare register match, and also when binary counter overflows.

8.6.1 Operation

■ 16-bit Standard PWM Output (Timer 7 and 8)

PWM waveform with an arbitrary duty is generated by setting a duty of PWM "High" period to TMnOC1 register. Its cycle is the time when 16-bit timer full count overflow.
Table:8.6.1 shows the PWM output pin.

Table:8.6.1 PWM Output Pin

Table remarks √: With function -: Without function

	PWM output pin	MN101EFA8	MN101EFA3	MN101EFA7	MN101EFA2
Timer 7	TM7IOA output	√	√	√	√
	TM7IOB output	√	√	√	√
Timer 8	TM8IOA output	√	√	√	√
	TM8IOB output	√	√	√	√

■ Count Timing of Standard PWM Output (at Normal) (Timer 7 and 8)

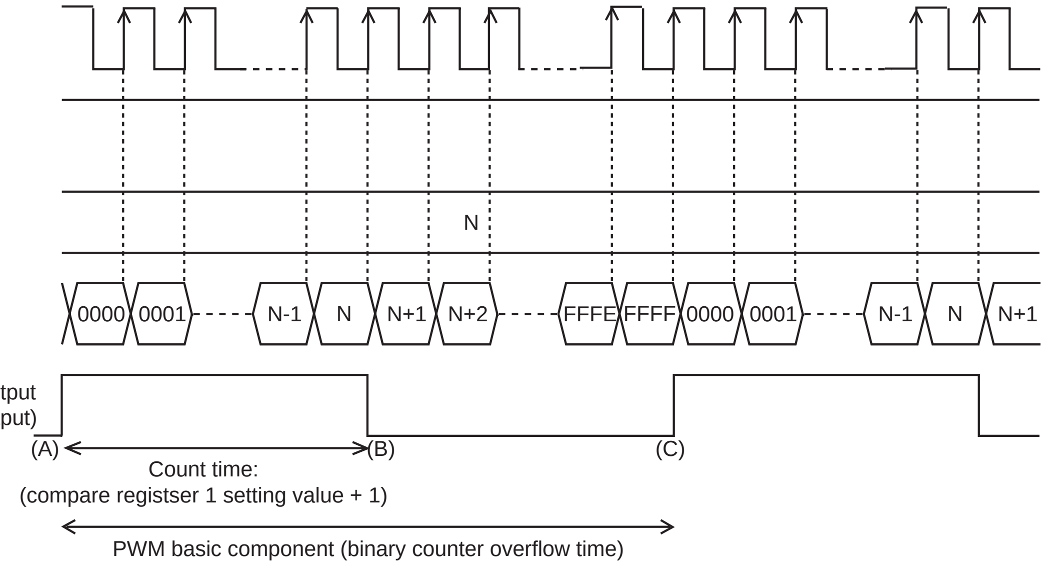


Figure:8.6.1 Count Timing of Standard PWM Output (at Normal)

Before starting the second PWM or later, clear binary counter and PWM waveform by writing to preset register as the PWM output waveform of the first cycle cannot be guaranteed.

- (A) shows "High" until binary counter reaches compare register value from 0x0000.
- (B) shows "Low" after the compare match, then binary counter counts up till the overflow.
- (C) shows "High" again if binary counter overflow.

■ Count Timing of Standard PWM Output (when compare register 1 is 0x0000) (Timer 7 and 8)

The count timing at setting 0x0000 to compare register 1 is shown below.

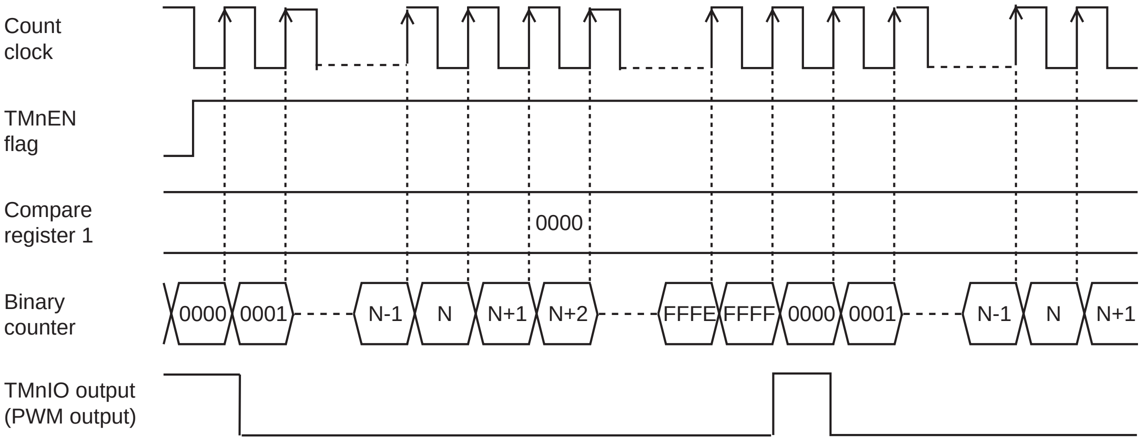


Figure:8.6.2 Count Timing of Standard PWM Output (when compare register 1 is 0x0000)

PWM output shows "High", when TM7EN flag is stopped (at "0").

■ Count Timing of Standard PWM Output (when compare register 1 is 0xFFFF) (Timer 7 and 8)

The count timing at setting 0xFFFF to compare register 1 is shown below.

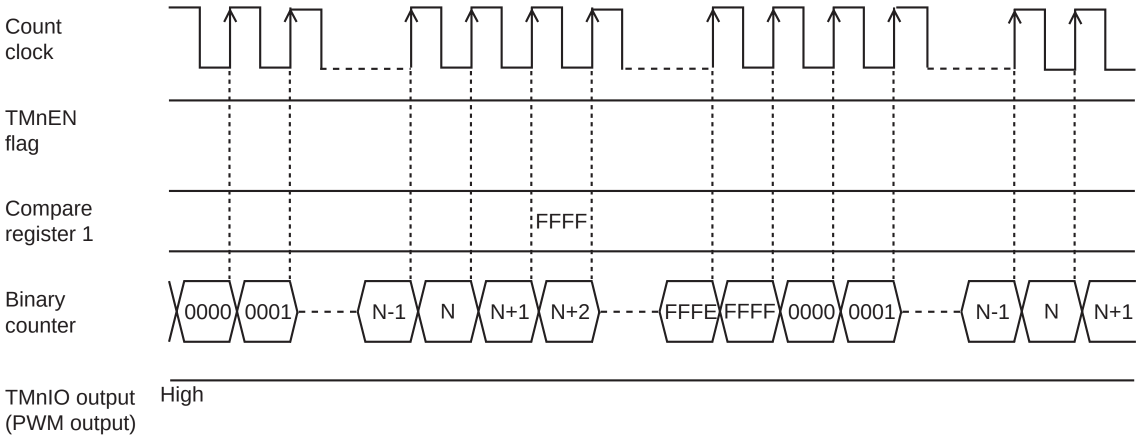


Figure:8.6.3 Count Timing of Standard PWM Output (when compare register 1 is 0xFFFF)

To execute the standard PWM output, set the TMnBCR flag of TMnMD2 register to “0” to select the full count overflow as the binary counter clear source and the PWM output set ("High" output) source.

The TMnOC1 compare match or the TMnOC2 compare match can be selected as a PWM output reset ("Low" output) source with the TnPWMSL flag of TMnMD2 register.

In the initial state of the PWM output, it is changed to "High" output from "Low" output at the timing that the PWM operation is selected by the TMnPWM flag of TMnMD2 register.

To guarantee the PWM waveform of the first cycle, after PWM operation is stopped, write to preset register to clear binary counter and the PWM waveform when restarting the PWM operation.

8.6.2 Setup Example

■ Standard PWM Output Setup Example

Pin TM7IOA outputs the 1/4 duty PWM output waveform at 152.6 Hz with Timer 7 (at the high speed oscillation, fpll-div = 10 MHz). One cycle of the PWM output waveform is decided by the overflow of binary counter. "High" period of the PWM output waveform is decided by the set value of compare register 1.
An example setup procedure, with a description of each step is shown below.

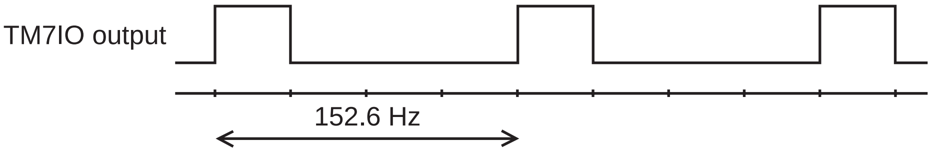


Figure:8.6.4 Output Waveform of TM7IO Output Pin

Setup Procedure	Description
(1) Stop the counter TM7MD1 (0x03F88) bp4: TM7EN =0	(1) Set the TM7EN flag of TM7MD1 register to “0” to stop Timer 7 counting.
(2) Set the special function pin to output PAOMD (0x03EBA) bp5: PAOMD5 =1 PADIR (0x03E9A) bp5: PADIR5 =1	(2) Set the PAOMD5 flag of PAOMD register to “1” to set PA5 pin as a special function pin. Set the PADIR5 flag of PADIR register to “1” to set the output mode. [Chapter 5 I/O Port]
(3) Set the PWM output TM7MD2 (0x03F89) bp4: TM7PWM =1	(3) Set the TM7PWM flag of TM7MD2 register to “1” to select the PWM output.
(4) Set the standard PWM output TM7MD2 (0x03F89) bp5: TM7BCR =0	(4) Set the TM7BCR flag of TM7MD2 register to “0” to select the full count overflow as the binary counter clear source.
(5) Select the count clock source TM7MD1 (0x03F88) bp1 to 0: TM7CK1 to 0 =00 bp3 to 2: TM7PS1 to 0 =00	(5) Select fpll-div as the clock source by the TM7CK1 to 0 flags of TM7MD1 register. Also, select 1/1 dividing as the count clock source by the TM7PS1 to 0 flags.
(6) Set "High" period of the PWM output TM7PR1(0x03F85, 0x03F84) =0x3FFF	(6) Set "High" period of the PWM output to TM7PR1 register. To set 1/4 duty of the full count 65536, set as; 65536/4-1=16383 (0x03FFF) At the same time, the same value is loaded to TM7OC1 register and TM7BC is initialized to 0x0000.
(7) Start the timer operation TM7MD1 (0x03F88) bp4: TM7EN =1	(7) Set the TM7EN flag of TM7MD1 register to “1” to operate Timer 7.

TM7BC counts up from 0x0000. The PWM source waveform outputs "High" until TM7BC reaches the set value of TM7OC1 register, then after the match it outputs "Low". After that, TM7BC continues to count up.
Once an overflow occurs, the PWM source waveform outputs "High" again, and TM7BC counts up from 0x0000, again.

8.7 16-bit High Precision PWM Output (Cycle/Duty can be changed consecutively)

Pin TMnIO outputs high precision PWM output, which is determined when the values of binary counter and compare register 1 match and when binary counter and compare register 2 match.

8.7.1 Operation

■ 16-bit High Precision PWM Output Operation (Timer 7 and 8)

The PWM waveform of any cycle/duty is generated by setting the cycle of PWM to TMnOC1 register and setting the duty of the "High" period to TMnOC2 register.

■ Count Timing of High Precision PWM Output (at Normal) (Timer 7 and 8)

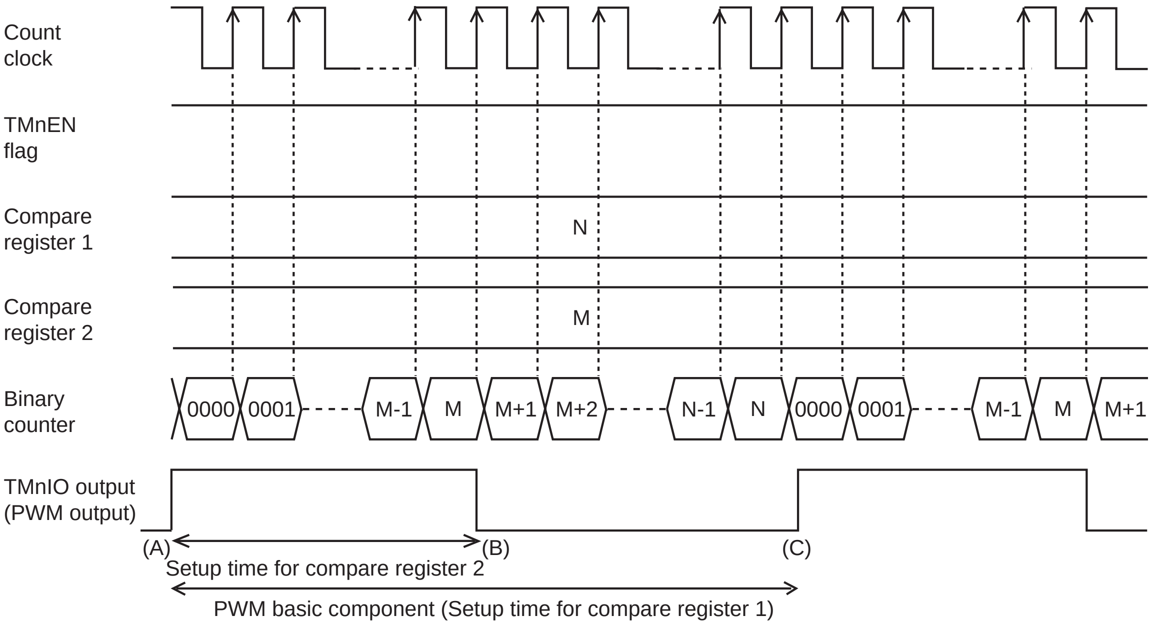


Figure:8.7.1 Count Timing of High Precision PWM Output (at Normal)

PWM output waveform,

(A) shows "High" until binary counter reaches compare register from 0x0000.

(B) shows "Low" after TMnOC2 compare match, binary counter then counts up until binary counter reaches TMnOC1 compare register is cleared.

(C) shows "High" again, when the binary counter is cleared.

■ Count Timing of High Precision PWM Output (When compare register 2 is 0x0000) (Timer 7 and 8)

Figure:8.7.2 shows count timing as compare register 2 is set to 0x0000.

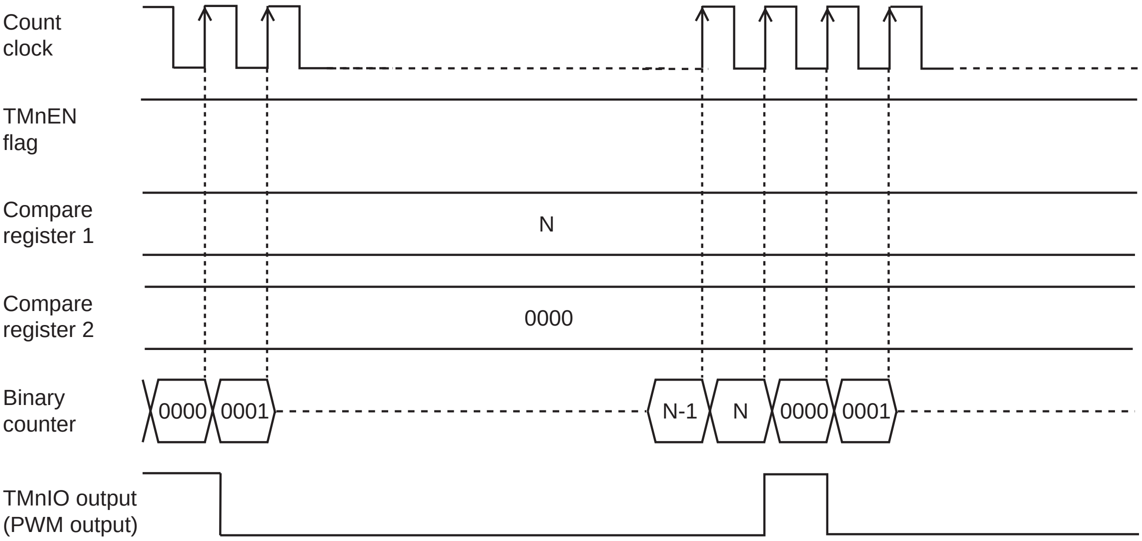


Figure:8.7.2 Count Timing of High Precision PWM Output (When compare register 2 is 0x0000)

When the TMnEN flag is stopped (at "0"), the PWM output shows "High".

■ Count Timing of High Precision PWM Output (When compare register 2 = the compere register 1 - 1) (Timer 7 and 8)

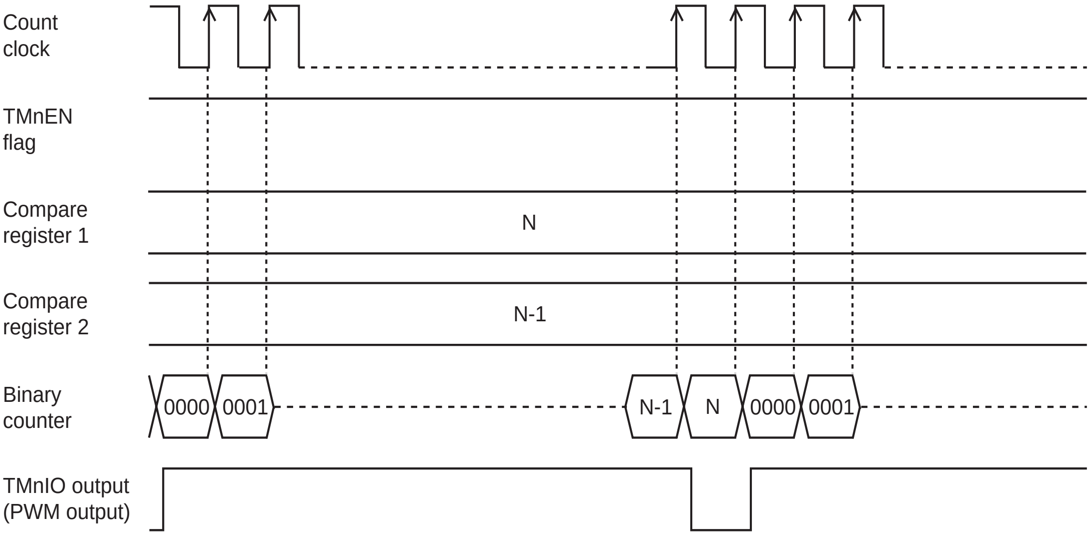


Figure:8.7.3 Count Timing of High Precision PWM Output (When compare register 2 = the compare register 1 - 1)

To execute the high precision PWM output, set the TMnBCR flag of TMnMD2 register to “1” to select TMnOC1 compare match as the clear source for binary counter, and the set ("High" output) source of the PWM output.
Also, set the TnPWMLS flag to “1” to select TMnOC2 compare match as the reset ("Low" output) source of the PWM output.

In the initial state of the PWM output, it is changed to "High" output from "Low" output at the timing that the PWM operation is selected by the TMnPWM flag of TMnMD2 register.

Set as the set value of TMnOC2 < the set value of TMnOC1.
If it is set as the set value of TMnOC2 ≥ the set value of TMnOC1, the PWM output is a "High" fixed output.

8.7.2 Setup Example

■ High Precision PWM Output Setup Example

Pin TM7IOA outputs the 1/4 duty PWM output waveform at 400 Hz with Timer 7.
Select fpll-div/2 (at fpll-div=10 MHz) as the clock source. One cycle of the PWM output waveform is decided by the set value of compare register 1. "High" period of the PWM output waveform is decided by the set value of compare register 2.
An example setup procedure, with a description of each step is shown below.

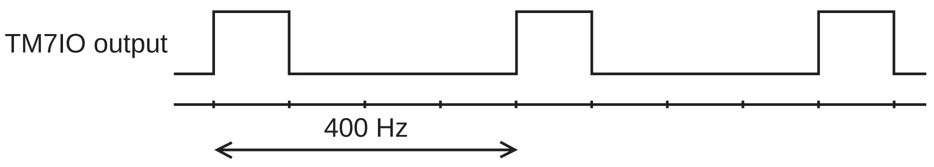


Figure:8.7.4 Output Waveform of Pin TM7IO output

Setup Procedure	Description
(1) Stop the counter TM7MD1(0x03F88) bp4: TM7EN =0	(1) Set the TM7EN flag of TM7MD1 register to “0” to stop Timer 7 counting.
(2) Set the special function pin to output PAOMD (0x03EBA) bp5: PAOMD5 =1 PADIR (0x03E9A) bp5: PADIR5 =1	(2) Set the PAOMD5 flag of PAOMD register to “1” to set PA5 pin as a special function pin. Set the PADIR5 flag of PADIR register to “1” to set the output mode. [Chapter 5 I/O Port]
(3) Set the PWM output TM7MD2(0x03F89) bp4: TM7PWM =1	(3) Set the TM7PWM flag of TM7MD2 register to “1” to select the PWM output.
(4) Set the high precision PWM output TM7MD2(0x03F89) bp5: TM7BCR =1 bp6: T7PWMSL =1	(4) Set the TM7BCR flag of TM7MD2 register to "1" to select TM7OC1 compare match as the clearing factor of the binary counter. In addition, set the T7PWMSL flag to "1" to select TM7OC2 compare match as the duty determination factor of PWM output.
(5) Select the count clock source TM7MD1(0x03F88) bp1 to 0: TM7CK1 to 0 =00 bp3 to 2: TM7PS1 to 0 =00	(5) Select fpll-div as the clock source by the TM7CK1 to 0 flags of TM7MD1 register. Also, select 1/1 dividing as the count clock source by theTM7PS1 to 0 flags.
(6) Set the PWM output cycle TM7PR1(0x03F85, 0x03F84) =0x61A7	(6) Set the PWM output cycle to TM7PR1 register. To set 400 Hz by dividing 10 MHz, set as; 25000-1=24999 (0x61A7). At the same time, the same value is loaded to TM7OC1 register, TM7BC is initialized to 0x0000.
(7) Set the "High" period of the PWM output TM7PR2(0x03F8D, 0x03F8C) =0x1869	(7) Set "High" period of the PWM output to TM7PR2 register. To set 1/4 duty of 25000 dividing, set as; 25000/4=6250 (0x1869) At the same time, the same value is loaded TM7OC2 register.
(8) Start the timer operation TM7MD1(0x03F88) bp4: TM7EN =1	(8) Set the TM7EN flag of TM7MD1 register to “1” to operate Timer 7.

TM7BC counts up from 0x0000. The PWM source waveform outputs "High" until TM7BC matches the set value of TM7OC2 register. Once they matches, it outputs "Low". After that, TM7BC continues to count up.
Once TM7BC matches TM7OC1 register to be cleared, the PWM output waveform outputs "High" again and TM7BC counts up from 0x0000 again.

8.8 16-bit Timer Capture

8.8.1 Operation

The value of binary counter is read during occurrence of the external interrupt input signal which is synchronized to fpll-div, fs or external event signal, Timer 0 or Timer 1 interrupt or the writing any value to the capture register.

- Capture Operation with External Interrupt Signal as the Trigger (Timer 7 and 8)

Input capture trigger is generated at the external interrupt input signal. The capture trigger is selected by TMnMD1 register and TMnMD2 register.
Selectable capture triggers and the interrupt flag setup are shown below.

Table:8.8.1 Capture Trigger

Capture trigger source	TMnMD2		TMnMD1
	TnICT1 to 0	TnICEDG0	TnICEDG1
IRQ0 falling edge	00 (IRQ0)	1	0
IRQ0 rising edge	00 (IRQ0)	1	1
IRQ0 both edges	00 (IRQ0)	0	X
IRQ1 falling edge	01 (IRQ1)	1	0
IRQ1 rising edge	01 (IRQ1)	1	1
IRQ1 both edges	01 (IRQ1)	0	X
IRQ2 falling edge	10 (IRQ2)	1	0
IRQ2 rising edge	10 (IRQ2)	1	1
IRQ2 both edges	10 (IRQ2)	0	X

If the system clock (fs) is selected as the capture clock and the capture operation is done during TMnIO input or operation with fpll-div, an incomplete value at the count up of binary counter may be written to input capture register.
To prevent this, use synchronous TMnIO input as the count clock.
[Chapter 8 8.4.1 Operation]

Capture trigger signal of the 16-bit Timer n is generated by sampling the rising edge of the capture clock selected by the TMnCKSMP flag of TMnMD3 register.
Therefore, even capture trigger is input, the value of the binary counter is not loaded to capture register until the rising edge of the next capture clock.
If the clock which is slower than CPU operation speed (fs) is set as the timer source clock, set the TMnCKSMP of TMnMD3 register to fs.
Also, the interval of each capture trigger should be set more than 2 cycles of the clock which is set at the TMnCKSMP of TMnMD3 register.

If the capture clock frequency is longer against the system clock, the value of capture register may be read out before capturing.

- Capture Count Timing as Both Edges of External Interrupt Signal are Selected as Trigger (Timer 7 and 8)

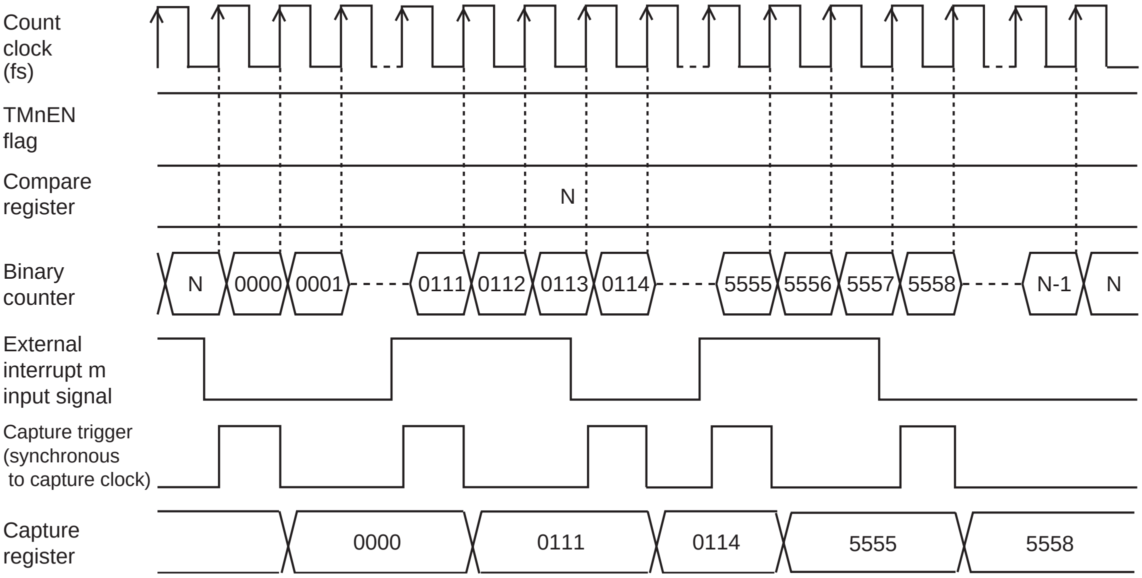


Figure:8.8.1 Capture Count Timing as External Interrupt Signal is selected as Trigger (Timer 7 and 8)

A capture trigger is generated at the both edges of external interrupt m input signal. In synchronized with this capture trigger, the value of binary counter is loaded to input capture register.
The value loaded to capture register is depending on the value of binary counter at the falling edge of the capture trigger. When the specified edge is selected as the capture trigger source, the capture trigger is generated only at that edge. The other count timing is the same as the count timing of the timer operation.

When binary counter is used as a free counter which counts 0x0000 to 0xFFFF set compare register 1 to 0xFFFF, or set the TMnBCR flag of TMnMD2 to “0”.

Even if an event is generated before the value of input capture register is read out, the value of input capture register can be rewritten.
If the writing and reading of the value of input capture register are operated at the same time, an uncompleted value may be read.

In the initial state after releasing the reset, the generation of trigger by the external interrupt signal is disabled. Set the TnICEN flag of TMnMD2 register to “1” to enable the trigger generation.

When using external interrupt signal as capture trigger, enable external interrupt input by IRQCNT register.

■ Capture Operation Triggered by Writing Software (Timer 7 and 8)

A capture trigger is generated by writing an arbitrary value to TMnIC register.
When writing a value to the register, the capture trigger is synchronized with the clock which is set by the TMnCKSMP flag of TMnMD3 register is generated. At the timing of the capture trigger falling, the value of binary counter can be loaded into input capture register.

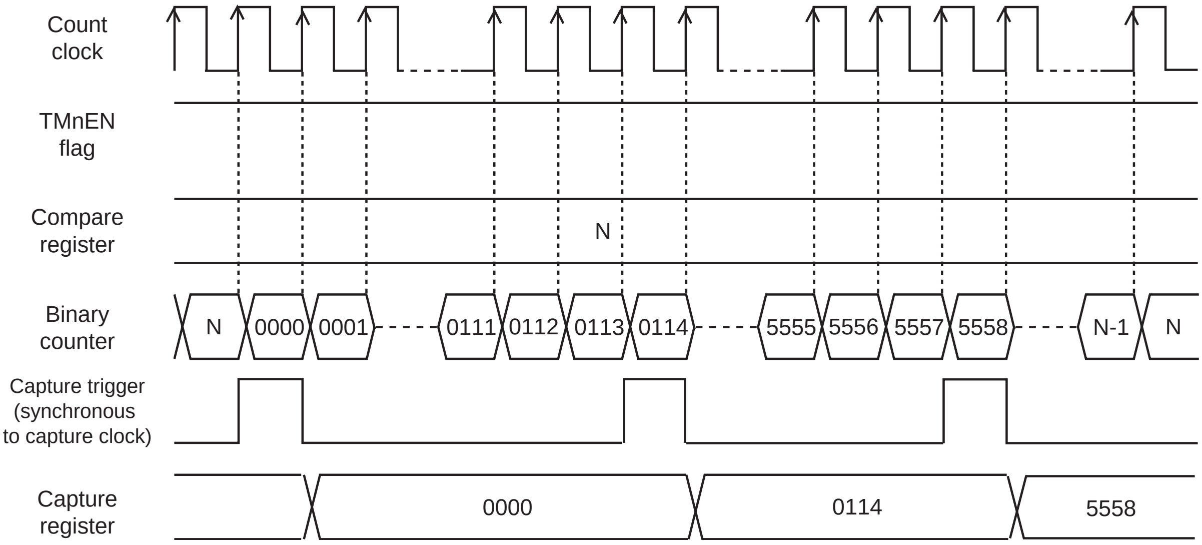


Figure:8.8.2 Capture Count Timing Triggered by Writing Software (Timer 7 and 8)

The capture trigger is generated at the writing signal to input capture register. The writing signal is generated at the last cycle of the write instruction. In synchronized with this capture trigger, the value of binary counter is loaded to input capture register. The value is depending on the value of binary counter at the falling edge of the capture trigger. The other timing is the same as the timer operation.

The writing to the input capture to generate the capture trigger should be done with 8-bit access instruction of TMnICL register or TMnICH register.
At this time, data is not actually written to TMnIC register.

On hardware, there is no flag to disable the capture operation triggered by writing software. Capture operation is enabled regardless of the TnICEN flag of TMnMD2 register.

■ Capture Operation as Timer 0 and Timer 1 Interrupts are Selected as Trigger (Timer 7 and 8)

A capture trigger of the input capture function is generated by Timer 0 and Timer 1 interrupt signals.
Select the capture trigger by TMnMD2 register and TMnMD4 register. When Timer 0 and Timer 1 interrupt signals are selected as the capture trigger, the edges of the capture trigger are disabled.

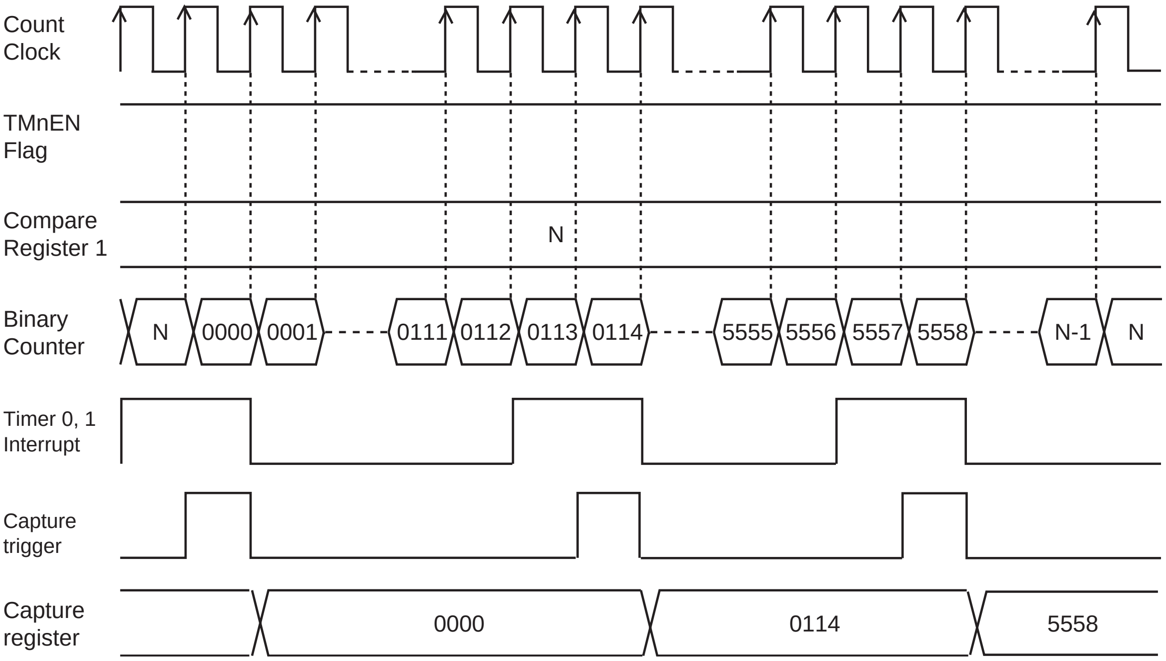


Figure:8.8.3 Capture Operation as Timer 0 and Timer 1 Interrupts are Selected as Trigger (Timer 7 and 8)

! When the TnICT1 to 0 flags of TMnMD2 register are set to "11", a capture trigger of the input capture function is generated by Timer 0 and Timer 1 interrupt signals. Select the capture trigger by TMnMD2 register and TMnMD4 register. When Timer 0 and Timer 1 interrupt signals are selected as the capture trigger, the edges of the capture trigger are disabled. When the TMnCKSMP flag is set to “Count clock” to execute the event count operation, Timer 0 and Timer 1 interrupt signals may not be recognized. To prevent this, select the synchronous TMnIO input as the clock source.

■ Binary Counter Clearance at the Timing of Capture (Timer 7 and 8)

When selecting the external interrupt input signal or Timer 0 and Timer 1 interrupts as the capture trigger, binary counter can be cleared during capture operation by setting the TnCAPCLR flag of TMnMD4 register to "1". Binary counter can only be cleared during timer count operation.

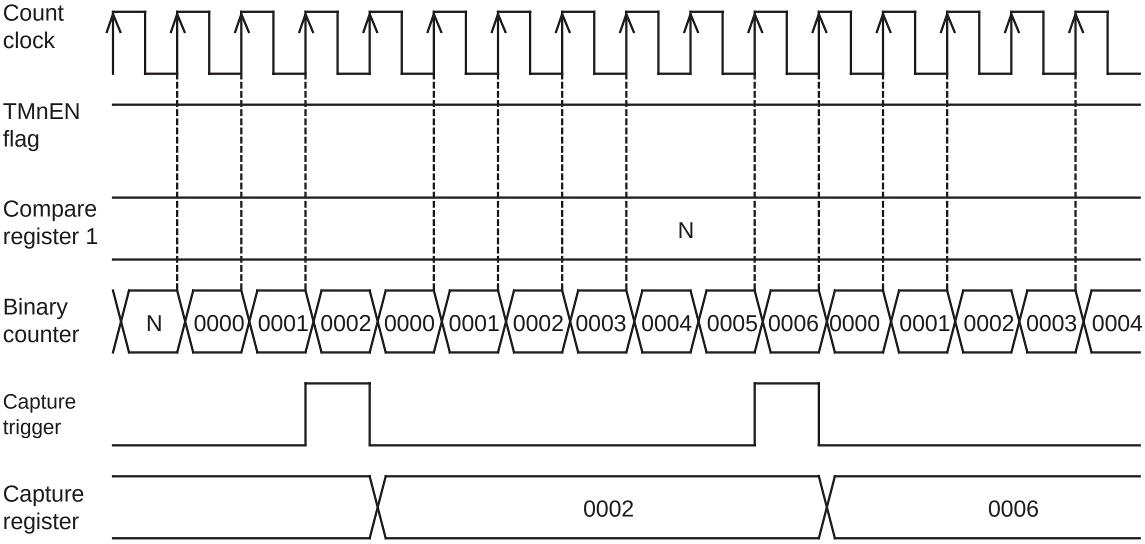


Figure:8.8.4 Binary Counter Clearance at the Timing of Capture (Timer 7 and 8)

8.8.2 Setup Example

■ Capture Function Setup Example

Pulse width measurement is enabled by storing the value of binary counter to capture register at the interrupt generation edge of external interrupt 0 signal with Timer 7. The interrupt generation edge is specified to be the rising edge.
An example setup procedure, with a description of each step is shown below.

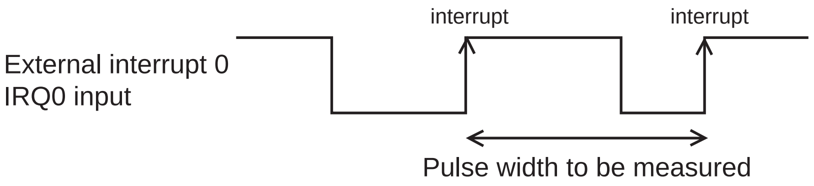


Figure:8.8.5 Pulse Width Measurement of External Interrupt 0 Input Signal

Setup Procedure	Description
(1) Stop the counter TM7MD1(0x03F88) bp4: TM7EN =0	(1) Set the TM7EN flag of TM7MD1 register to “0” to stop Timer 7 counting.
(2) Disable the interrupt IRQ0ICR(0x03FE2) bp1: IRQ0IE =0	(2) Set the IRQIE flag of IRQ0ICR register to “0” to disable the interrupt.
(3) Select the timer clear source TM7MD2(0x03F89) bp5: TM7BCR =1	(3) Set the TM7BCR flag of TM7MD2 register to “1” to select the compare match as the binary counter clear source.
(4) Select the count clock source TM7MD1(0x03F88) bp1 to 0: TM7CK1 to 0 =00 bp3 to 2: TM7PS1 to 0 =00	(4) Select fpll-div as the clock source by the TM7CK1 to 0 flags of TM7MD1 register. Also, select 1/1 dividing of fpll-div as the count clock source by the TM7PS1 to 0 flags.
(5) Set the compare register TM7PR1(0x03F85, 0x03F84) =0xFFFF	(5) Set 0xFFFF to TM7PR1 register. At that time, the same value is loaded to TM7OC1 register, TM7BC is initialized to 0x0000.
(6) Select the capture trigger generation interrupt source TM7MD2(0x03F89) bp1 to 0: T7ICT1 to 0 =00	(6) Select external interrupt 0 (IRQ0) input as the capture trigger generation source by the T7ICT1 to 0 flags of TM7MD2 register.
(7) Select the capture trigger generation edge TM7MD1(0x03F88) bp6: T7ICEDG1 =1 TM7MD2 (0x03F89) bp7: T7ICEDG0 =1	(7) Set the T7ICEDG1 flag of TM7MD1 register to “1” to select the rising edge as the capture trigger generation edge. Also, set the T7ICEDG0 flag of TM7MD2 register to “1” to enable the specify edge as the capture trigger generation source.
(8) Select the capture sampling TM7MD3(0x03F9E) bp7: TM7CKSMP =0	(8) Select the capture sampling as the count clock.
(9) Select the interrupt generation valid edge IRQ0ICR(0x03FE2) bp5: REDG0 =1	(9) Set the REDG0 flag of IRQ0ICR register to “1” to select the rising edge as the interrupt generation valid edge.
(10) Set the interrupt level IRQ0ICR(0x03FE2) bp7 to 6: IRQ0LV1 to 0 =10	(10) Set the interrupt level by the IRQ0LV1 to 0 flags of IRQ0ICR register. If the interrupt request flag has already been set, clear the request flag. [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup]
(11) Enable the interrupt IRQ0ICR(0x03FE2) bp1: IRQ0IE =1	(11) Set the IRQ0IE flag of IRQ0ICR register to “1” to enable the interrupt.
(12) Enable the capture trigger generation TM7MD2(0x03F89) bp2: T7ICEN =1	(12) Set the T7ICEN flag of TM7MD2 register to “1” to enable the capture trigger generation.
(13) Start the timer operation TM7MD1(0x03F88) bp4: TM7EN =1	(13) Set the TM7EN flag of TM7MD1 register to “1” to operate Timer 7.

TM7BC counts up from 0x0000. At the timing of the rising edge of external interrupt 0 input signal, the value of TM7BC is loaded to TM7IC register. At that time, the pulse width between rising edge of the external interrupt input signal can be measured by reading the value of TM7IC register through interrupt service routine, and calculating the difference between the capture values.

Chapter 9　Motor Control 16-bit TImer

9.1 Overview

This LSI contains a motor control 16-bit timer function (timer 9).
In addition to the normal 16-bit timer function, Timer 9 of this LSI can be used as complementary 3 phases PWM and 4 phases PWM for motor control applications.

Table:9.1.1 shows the list of Timer 9 used pins. The used pins can be switched between A type or B type. For A type, "A" is added to the end of the pin name, and for B type, "B" is added.

Table:9.1.1 Timer 9 Used Pins

Table remarks √: With function -: Without function

Functions	Pin Name	MN101EFA8	MN101EFA3	MN101EFA7	MN101EFA2
TM9IOA	PA7	√	√	√	√
TM9IOB	P01	√	√	√	√
TM9OD0	P80	√	√	√	√
TM9OD1	P81	√	√	√	√
TM9OD2	P82	√	√	√	√
TM9OD3	P83	√	√	√	√
TM9OD4	P84	√	√	√	√
TM9OD5	P85	√	√	√	√

	Timer 9
16-bit timer function	TM9IOA
	TM9IOB
Complementary 3 phases PWM function	TM9OD0
	TM9OD1
	TM9OD2
	TM9OD3
	TM9OD4
	TM9OD5
4 phases PWM function	TM9OD0 First phase
	TM9OD2 Second phase
	TM9OD4 Third phase
	TM9OD1 Forth phase

In this manual, if there is not much difference in the function between Pin A and B, “A” and “B” of the pin names are omitted.

When changing fpll-div frequency by bp7 to bp4 of OSCCNT register, it should be executed after the motor control 16-bit timer is stopped.

9.1.1 Functions

Table:9.1.2 Motor Control 16-bit Timer Functions

		Timer 9 (16-bit timer for motor control)
16-bit timer	interrupt factor	PWMOVIRQ, PWMUDIRQ, TM9OC2IRQ
	Timer operation	√
	Event count	TM9IOA input/ TM9IOB input
	Timer pulse output	TM9IOA output/ TM9IOB output
Complementary 3 phases PWM	interrupt factor	PWMOVIRQ, PWMUDIRQ, TM9OC2IRQ
	Complementary 3 phases PWM output	U-phase, V-phase, W-phase
	Waveform mode	Triangular wave, Saw-tooth wave
	Dead Time setting	√
	High/Low level output	√
4 phases PWM	interrupt factor	PWMOVIRQ, PWMUDIRQ
	4 phases PWM output	Stepping motor 1-phase excitation 2-phase excitation 1-phase to 2-phase excitation PWM output
	Waveform mode	Triangular wave, Saw-tooth wave
	Dead Time setting	√
	Output order inversion	√
Clock source		fpll-div
		fpll-div/2
		fpll-div/4
		fpll-div/16
		fs
		fs/2
		fs/4
		fs/16
		Synchronous TM9IO input
		Synchronous TM9IO input/2
		Synchronous TM9IO input/4
		Synchronous TM9IO input/16

fpll-div: High-speed clock
fs: System clock [Chapter 3 Clock Control]

9.1.2 Block Diagram

■ Motor Control PWM Block Diagram

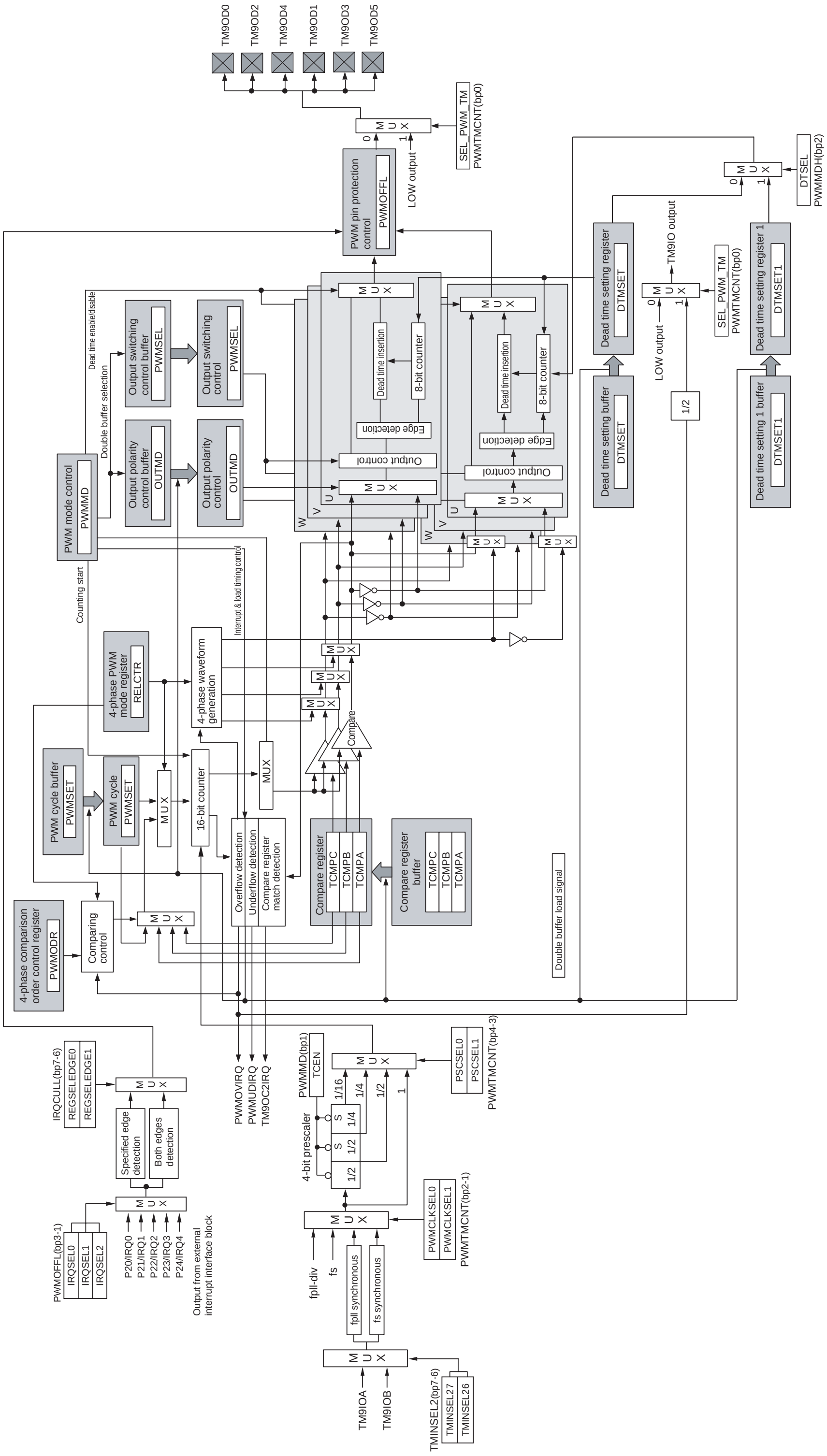


Figure:9.1.1 16-bit TImer for Motor Control Block Diagram

9.2 Control Registers

9.2.1 Control Registers for Motor Control 16-bit Timer

Table:9.2.1 shows registers which controls motor control 16-bit timer applications.

Table:9.2.1 Control Registers for Motor Control PWM

Table remarks √: With function -: Without function

Register	Address	R/W	Function	Page	MN101 EFA8	MN101 EFA3	MN101 EFA7	MN101 EFA2
PWMMDL	0x03E00	R/W	PWM mode control register lower 8 bits	IX-8	√	√	√	√
PWMMDH	0x03E01	R/W	PWM mode control register upper 8 bits	IX-9	√	√	√	√
PWMSELL	0x03E02	R/W	PWM output control register lower 8 bits	IX-11	√	√	√	√
PWMSELH	0x03E03	R/W	PWM output control register upper 8 bits	IX-12	√	√	√	√
PWMSETL	0x03E04	R/W	PWM cycle setting register lower 8 bits	IX-13	√	√	√	√
PWMSETH	0x03E05	R/W	PWM cycle setting register upper 8 bits	IX-13	√	√	√	√
TCMPAL	0x03E06	R/W	PWM U-phase comparison setting register lower 8 bits	IX-14	√	√	√	√
TCMPAH	0x03E07	R/W	PWM U-phase comparison setting register upper 8 bits	IX-14	√	√	√	√
TCMPBL	0x03E08	R/W	PWM V-phase comparison setting register lower 8 bits	IX-14	√	√	√	√
TCMPBH	0x03E09	R/W	PWM V-phase comparison setting register upper 8 bits	IX-15	√	√	√	√
TCMPCL	0x03E0A	R/W	PWM W-phase comparison setting register lower 8 bits	IX-15	√	√	√	√
TCMPCH	0x03E0B	R/W	PWM W-phase comparison setting register upper 8 bits	IX-15	√	√	√	√
OUTMD	0x03E0C	R/W	PWM output polarity control register	IX-10	√	√	√	√
DTMSET	0x03E0D	R/W	PWM dead time setting register	IX-17	√	√	√	√
DTMSET1	0x03E0E	R/W	PWM dead time setting register 1	IX-17	√	√	√	√
PWMBCL	0x03E10	R	PWM BC value read lower 8 bits	IX-18	√	√	√	√
PWMBCH	0x03E11	R	PWM BC value read upper 8 bits	IX-18	√	√	√	√
BCSTR	0x03E12	R	PWM BC status read register	IX-19	√	√	√	√
PWMOFFL	0x03E13	R/W	PWM pin protection control register lower 8 bits	IX-20	√	√	√	√
PWMOFFH	0x03E14	R/W	PWM pin protection control register upper 8 bits	IX-21	√	√	√	√
IRQCULL	0x03E15	R/W	PWM interrupt output control register	IX-22	√	√	√	√
PWMTMCNT	0x03E16	R/W	PWM timer operation control register	IX-23	√	√	√	√
RELCTR	0x03E17	R/W	4 phases PWM mode register	IX-24	√	√	√	√
PWMODR	0x03E18	R/W	4 phases PWM output order control register	IX-25	√	√	√	√
RELSTAT	0x03E19	R	4 phases PWM output status register	IX-26	√	√	√	√
PWMCMP1	0x03E1A	W	4 phases PWM cycle/comparison collective setting register 1	IX-27	√	√	√	√
PWMCMP2	0x03E1B	W	4 phases PWM cycle/comparison collective setting register 2	IX-28	√	√	√	√

Table remarks √: With function -: Without function

Register	Address	R/W	Function	Page	MN101 EFA8	MN101 EFA3	MN101 EFA7	MN101 EFA2
PWMOVICR	0x03FF9	R/W	Timer 9 overflow interrupt control register	IV-26	√	√	√	√
PWMUDICR	0x03FFA	R/W	Timer 9 underflow interrupt control register	IV-27	√	√	√	√
PERIILR	0x03FFE	R/W	Peripheral function group interrupt control register	IV-35	√	√	√	√
IRQEXPEN	0x03F4E	R/W	Peripheral function group input enable register	IV-36	√	√	√	√
IRQEXPDT	0x03F4F	R/W	Peripheral function group interrupt factor retention register	IV-37	√	√	√	√
TMCKSEL2	0x03FB1	R/W	Timer clock selection register 2	VIII-15	√	√	√	√
TMINSEL2	0x03FB3	R/W	Timer input selection register 2	VIII-16	√	√	√	√

R/W: Readable / Writable
R: Read only
W: Write only

9.2.2 PWM Mode Control Register

PWM mode control registers are used to set various modes for Timer 9.

- PWM Mode Control Register Lower 8 bits (PWMMDL: 0x03E00)

bp	7	6	5	4	3	2	1	0
Flag	PCRAEN	PCRBEN	INTAEN	INTBEN	DTEN	ORMD	TCEN	WAVEMD
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7	PCRAEN	Double buffer mode load timing enable (PWM binary counter underflow) 0: Disabled 1: Enabled
6	PCRBEN	Double buffer mode load timing enable (PWM binary counter overflow) 0: Disabled 1: Enabled
5	INTAEN	Timer 9 underflow interrupt (PWMUDIRQ) enable 0: Disabled 1: Enabled
4	INTBEN	Timer 9 overflow interrupt (PWMOVIRQ) enable 0: Disabled 1: Enabled
3	DTEN	Dead Time insertion 0: Not inserted 1: Inserted
2	ORMD	Dead Time insertion logic 0: Positive logic (High active) 1: Negative logic (Low active)
1	TCEN	PWM counting operation enable 0: Disabled 1: Enabled
0	WAVEMD	PWM waveform mode 0: Triangular wave 1: Saw-tooth wave

- PWM Mode Control Register Upper 8 bits (PWMMDH: 0x03E01)

bp	7	6	5	4	3	2	1	0
Flag	-	INTCEN	-	Reserved	-	DTSEL	SDSELA	SDSELB
At reset	-	0	-	0	-	0	0	0
Access	-	R/W	-	R/W	-	W	R/W	R/W

bp	Flag	Description
7	-	-
6	INTCEN	Timer 9 TCMPA compare match interrupt (TM9OC2IRQ) enable 0: Disabled 1: Enabled
5	-	-
4	Reserved	Always set to “0”.
3	-	-
2	DTSEL	Dead time insertion mode selection 0: Insert same dead time value in all PWM outputs (The value of DTMSET register is valid only.) 1: Insert different dead time value in PWM output of TM9OD0,2,4 and TM9OD1,3,5. (The value of DTMSET, DTMSET1 register is valid only.)
1	SDSELA	OUTMD buffer mode 0: Single-buffer mode 1: Double-buffer mode
0	SDSELB	PWMSEL buffer mode 0: Single-buffer mode 1: Double-buffer mode

When using Timer 9 as a 16-bit timer, set the WAVEMD flag to “1” to select saw-tooth wave.

To set PWM mode control register, set the TCEN flag of PWMMDL register to "0" when the counting is stopped.

The DTSEL flag can only write it. The value of DTSEL flag is read to be “0”.

9.2.3 PWM Output Polarity Control Register

PWM output polarity control register is used to select polarity for each of the PWM outputs.

This register can select double-buffer or single-buffer mode by the SDSELA flag of PWM mode control register (PWMMDL/H). When double-buffer mode is selected, the value of OUTMD is loaded into the register at the timing selected with PWMMDL/H register. When the PWM counter is not running, the double-buffer value is loaded into the register as is regardless of the specified read timing; thus, ensuring smooth use of double buffer from the initial state where the PWM counter starts.

■ PWM Output Polarity Control Register (OUTMD: 0x03E0C)

bp	7	6	5	4	3	2	1	0
Flag	-	-	PXDTNW	PXDTW	PXDTNV	PXDTV	PXDTNU	PXDTU
At reset	-	-	0	0	0	0	0	0
Access	-	-	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-6	-	-
5	PXDTNW	Output polarity for TM9OD5 0: Negative phase 1: Positive phase
4	PXDTW	Output polarity for TM9OD4 0: Positive phase 1: Negative phase
3	PXDTNV	Output polarity for TM9OD3 0: Negative phase 1: Positive phase
2	PXDTV	Output polarity for TM9OD2 0: Positive phase 1: Negative phase
1	PXDTNU	Output polarity for TM9OD1 0: Negative phase 1: Positive phase
0	PXDTU	Output polarity for TM9OD0 0: Positive phase 1: Negative phase

9.2.4 PWM Output Control Register

PWM output control register is used to switch between 2 output sources, PWM output or High/Low level output.

This register can select double-buffer or single-buffer mode by the SDSELB flag of PWM mode control register (PWMMDL/H). When double-buffer mode is selected, the value of PWMSELL/H is loaded into the register at the timing selected with PWMMDL/H register. When the PWM counter is not running, the double-buffer value is loaded into the register as is regardless of the specified read timing.

- PWM Output Control Register Lower 8 bits (PWMSELL: 0x03E02)

bp	7	6	5	4	3	2	1	0
Flag	PSELN00	PSEL00	OTLVN02	OTLV02	OTLVN01	OTLV01	OTLVN00	OTLV00
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7	PSELN00	TM9OD1 output source 0: PWM output 1: High/Low level output
6	PSEL00	TM9OD0 output source 0: PWM output 1: High/Low level output
5	OTLVN02	TM9OD5 High/Low level output 0: Low level output 1: High level output
4	OTLV02	TM9OD4 High/Low level output 0: Low level output 1: High level output
3	OTLVN01	TM9OD3 High/Low level output 0: Low level output 1: High level output
2	OTLV01	TM9OD2 High/Low level output 0: Low level output 1: High level output
1	OTLVN00	TM9OD1 High/Low level output 0: Low level output 1: High level output
0	OTLV00	TM9OD0 High/Low level output 0: Low level output 1: High level output

■ PWM Output Control Register Upper 8 bits (PWMSELH: 0x03E03)

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	PSELN02	PSEL02	PSELN01	PSEL01
At reset	-	-	-	-	0	0	0	0
Access	-	-	-	-	R/W	R/W	R/W	R/W

bp	Flag	Description
7-4	-	-
3	PSELN02	TM9OD5 output source 0: PWM output 1: High/Low level output
2	PSEL02	TM9OD4 output source 0: PWM output 1: High/Low level output
1	PSELN01	TM9OD3 output source 0: PWM output 1: High/Low level output
0	PSEL01	TM9OD2 output source 0: PWM output 1: High/Low level output

9.2.5 PWM Cycle Setting Register

PWM cycle setting register is used to determine the cycle of 16-bit timer and PWM output.

This register needs to be set only when double-buffer mode is selected. The value of PWMSETL/H is loaded into the register at the timing selected with PWM mode control register (PWMMDL/H). When the PWM counter is not running, the double-buffer value is loaded into the register as is regardless of the specified read timing.

■ PWM Cycle Setting Register Lower 8 bits (PWMSETL: 0x0x03E04)

bp	7	6	5	4	3	2	1	0
Flag	PMSET07	PMSET06	PMSET05	PMSET04	PMSET03	PMSET02	PMSET01	PMSET00
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-0	PMSET07 to PMSET00	PWM cycle setting (except 4 phases PWM) First phase output period (4 phases PWM) Set the cycle in the PWM 16-bit counter.

■ PWM Cycle Setting Register Upper 8 bits (PWMSETH: 0x0x03E05)

bp	7	6	5	4	3	2	1	0
Flag	PMSET0F	PMSET0E	PMSET0D	PMSET0C	PMSET0B	PMSET0A	PMSET09	PMSET08
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-0	PMSET0F to PMSET08	PWM cycle setting (except 4 phases PWM) First phase output period (4 phases PWM) Set the cycle in the PWM 16-bit counter.

When accessing PWMSETL and PWMSETH registers, make sure to access the lower 8 bits (PWMSETL register) by the MOVW instruction.

9.2.6 PWM Phase Comparison Setting Register

PWM phase comparison setting register is used to determine the timing when the outputs of 3 phases(U/V/W) of PWM change. This register needs to be set only when double-buffer mode is selected. The value of TCMPn is loaded into the register at the timing selected with PWM mode control register (PWMMDL/H). When the PWM counter is not running, the double-buffer value is loaded into the register as is regardless of the specified read timing.

■ PWM U-Phase Comparison Setting Register Lower 8 bits (TCMPAL: 0x03E06)

bp	7	6	5	4	3	2	1	0
Flag	TCPA07	TCPA06	TCPA05	TCPA04	TCPA03	TCPA02	TCPA01	TCPA00
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-0	TCPA07 to TCPA00	Timing setting for PWM U-phase output change (complementary 3 phases) Second phase output period (4 phases PWM) Set the value of PWM U-phase to compare with binary counter

■ PWM U-Phase Comparison Setting Register Upper 8 bits (TCMPAH: 0x03E07)

bp	7	6	5	4	3	2	1	0
Flag	TCPA0F	TCPA0E	TCPA0D	TCPA0C	TCPA0B	TCPA0A	TCPA09	TCPA08
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-0	TCPA0F to TCPA08	Timing setting for PWM U-phase output change (complementary 3 phases) Second phase output period (4 phases PWM) Set the value of PWM U-phase to compare with binary counter

■ PWM V-Phase Comparison Setting Register Lower 8 bits (TCMPBL: 0x03E08)

bp	7	6	5	4	3	2	1	0
Flag	TCPB07	TCPB06	TCPB05	TCPB04	TCPB03	TCPB02	TCPB01	TCPB00
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-0	TCPB07 to TCPB00	Timing setting for PWM V-phase output change (complementary 3 phases) Third phase output period (4 phases PWM) Set the value of PWM V-phase to compare with binary counter

- PWM V-Phase Comparison Setting Register Upper 8 bits (TCMPBH: 0x03E09)

bp	7	6	5	4	3	2	1	0
Flag	TCPB0F	TCPB0E	TCPB0D	TCPB0C	TCPB0B	TCPB0A	TCPB09	TCPB08
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-0	TCPB0F to TCPB08	Timing setting for PWM V-phase output change (complementary 3 phases) Third phase output period (4 phases PWM) Set the value of PWM V-phase to compare with binary counter

- PWM W-Phase Comparison Setting Register Lower 8 bits (TCMPCL: 0x03E0A)

bp	7	6	5	4	3	2	1	0
Flag	TCPC07	TCPC06	TCPC05	TCPC04	TCPC03	TCPC02	TCPC01	TCPC00
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-0	TCPC07 to TCPC00	Timing setting for PWM W-phase output change (complementary 3 phases) Forth phase output period (4 phases PWM) Set the value of PWM W-phase to compare with binary counter

- PWM W-Phase Comparison Setting Register Upper 8 bits (TCMPCH: 0x03E0B)

bp	7	6	5	4	3	2	1	0
Flag	TCPC0F	TCPC0E	TCPC0D	TCPC0C	TCPC0B	TCPC0A	TCPC09	TCPC08
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-0	TCPC0F to TCPC08	Timing setting for PWM W-phase output change (complementary 3 phases) Forth phase output period (4 phases PWM) Set the value of PWM W-phase to compare with binary counter

When accessing TCMPA, TCMPB and TCMPC registers, make sure to access to the lower 8 bits (TCMPnL register) by the MOVW instruction.

Each PWM phase comparison setting register should be set under the setting value of PWM cycle setting register.

9.2.7 PWM Dead Time Setting Register

Dead time setting register is used to set dead time of PWM output. Dead time is designed to insert an ON-time delay into each of the upper and lower phases when the signal is inverted at PWM output. The dead time counter functions in synchronization with clock set by PWM mode control register (PWMMD) and counts 1 every 2 clock cycles. Delay time of the dead time is calculated based on "setting × 2 + 1". Thus, when "00" is set, 1 clock cycle of dead time is inserted if dead time is enabled. This register needs to be set only when double-buffer mode is selected. The value of DTMSET is loaded into the register at the timing selected with PWM mode control register (PWMMD). When the PWM counter is not running, the double-buffer value is loaded into the register as is regardless of the specified read timing. Also, different dead times can be set in TM9OD0, 2, 4, and TM9OD1, 3, 5 by the DTSEL flag of PWM mode control register (PWMMDL/H).

- PWM Dead Time Setting Register (DTMSET: 0x03E0D)

bp	7	6	5	4	3	2	1	0
Flag	DTST07	DTST06	DTST05	DTST04	DTST03	DTST02	DTST01	DTST00
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-0	DTST07 to DTST00	PWM dead time setting When DTSEL="0" Set the dead time value to the 8-bit dead time counter. When DTSEL="1" Set the dead time value to the PWM output of TM9OD0, 2, 4.

- PWM Dead Time Setting Register 1 (DTMSET1: 0x03E0E)

bp	7	6	5	4	3	2	1	0
Flag	DTST17	DTST16	DTST15	DTST14	DTST13	DTST12	DTST11	DTST10
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-0	DTST17 to DTST10	PWM dead time setting When DTSEL="1" Set the dead time value to the PWM output of TM9OD1, 3, 5.

9.2.8 PWM BC Value Read Register

BC value read register is used to read binary counter value of Timer 9.

■ PWM BC Value Read Register Lower 8 bits (PWMBCL: 0x03E10)

bp	7	6	5	4	3	2	1	0
Flag	PWMBC07	PWMBC06	PWMBC05	PWMBC04	PWMBC03	PWMBC02	PWMBC01	PWMBC00
At reset	0	0	0	0	0	0	0	0
Access	R	R	R	R	R	R	R	R

bp	Flag	Description
7-0	PWMBC07 to PWMBC00	PWM binary counter value read Read binary counter value of PWM

■ PWM BC Value Read Register Upper 8 bits (PWMBCH: 0x03E11)

bp	7	6	5	4	3	2	1	0
Flag	PWMBC15	PWMBC14	PWMBC13	PWMBC12	PWMBC11	PWMBC10	PWMBC09	PWMBC08
At reset	0	0	0	0	0	0	0	0
Access	R	R	R	R	R	R	R	R

bp	Flag	Description
7-0	PWMBC15 to PWMBC08	PWM binary counter value read Read binary counter value of PWM

When the CPU reads binary counter, the read data is handled in 8-bits units in the LSI even if it is a 16-bit MOVW instruction. Thus, it will read the data incorrectly if a carry from the lower 8 bits to the upper 8 bits occurs during counting operation.

9.2.9 PWM BC Status Read Register

BC status read register is used to read binary counter's counting status of Timer 9.

■ PWM BC Status Read Register (BCSTR: 0x03E12)

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	-	-	-	PWMSTR
At reset	-	-	-	-	-	-	-	1
Access	-	-	-	-	-	-	-	R

bp	Flag	Description
7-1	-	-
0	PWMSTR	PWM binary counter’s counting status read 0: Down-count 1: Up-count

9.2.10 PWM Pin Protection Control Register

This register is used to automatically bring the PWM output pins into Hi-Z state or inactive output by the specified external interrupt generation.

■ PWM Pin Protection Control Register Lower 8 bits (PWMOFFL: 0x03E13)

bp	7	6	5	4	3	2	1	0
Flag	PRTANU1	PRTANU0	PRTAU1	PRTAU0	IRQSEL2	IRQSEL1	IRQSEL0	OUTEN0
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-6	PRTANU1 PRTANU0	TM9OD1 output protection function 00: Unused 01: Hi-Z output 10: Inactive output 11: Setting prohibited
5-4	PRTAU1 PRTAU0	TM9OD0 output protection function 00: Unused 01: Hi-Z output 10: Inactive output 11: Setting prohibited
3-1	IRQSEL2 IRQSEL1 IRQSEL0	PWM pin protection external interrupt selection 000: IRQ00 001: IRQ01 010: IRQ02 011: IRQ03 100: IRQ04 101: Setting prohibited 110: Setting prohibited 111: Setting prohibited
0	OUTEN0	Hi-Z output enable 0: Disabled 1: Enabled

Be sure to set the IRQSEL2 to IRQSEL0 flags before setting the other flags in PWMOFFL register.

■ PWM Pin Protection Control Register Upper 8 bits (PWMOFFH: 0x03E14)

bp	7	6	5	4	3	2	1	0
Flag	PRTANW1	PRTANW0	PRTAW1	PRTAW0	PRTANV1	PRTANV0	PRTAV1	PRTAV0
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-6	PRTANW1 PRTANW0	TM9OD5 output protection function 00: Unused 01: Hi-Z output 10: Inactive output 11: Setting prohibited
5-4	PRTAW1 PRTAW0	TM9OD4 output protection function 00: Unused 01: Hi-Z output 10: Inactive output 11: Setting prohibited
3-2	PRTANV1 PRTANV0	TM9OD3 output protection function 00: Unused 01: Hi-Z output 10: Inactive output 11: Setting prohibited
1-0	PRTAV1 PRTAV0	TM9OD2 output protection function 00: Unused 01: Hi-Z output 10: Inactive output 11: Setting prohibited

When the PWM output pin is set to Hi-Z output by the PRTAxx flag of PWMOFF register, the OUTEN0 flag should be set to “1”. If the OUTEN0 flag is not set to “1”, Hi-Z output is not enabled.

9.2.11 Interrupt Output Control Register

This register is used to control the interrupt factor signals PWMOVIRQ, PWMUDIRQ and TM9OC2IRQ, and also used to select edge of PWM pin protection function interrupt factor signals.

■ Interrupt Output Control Register (IRQCULL: 0x03E15)

bp	7	6	5	4	3	2	1	0
Flag	REGSEL EDGE1	REGSEL EDGE0	REGCUL LCMP1	REGCUL LCMP0	REGCUL LUDF1	REGCUL LUDF0	REGCUL LOVF1	REGCUL LOVF0
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-6	REGSELEDGE1 REGSELEDGE0	PWMOFF interrupt factor input edge selection 00: Both edges detected 01: Both edges detected 10: Falling edge detected 11: Rising edge detected
5-4	REGCULLCMP1 REGCULLCMP0	TM9OC2IRQ output control 00: Normal output 01: 4-for-1 output 10: 2-for-1 output 11: Setting prohibited
3-2	REGCULLUDF1 REGCULLUDF0	PWMUDIRQ output control 00: Normal output 01: 4-for-1 output 10: 2-for-1 output 11: Setting prohibited
1-0	REGCULLOVF1 REGCULLOVF0	PWMOVIRQ output control 00: Normal output 01: 4-for-1 output 10: 2-for-1 output 11: Setting prohibited

9.2.12 PWM Timer Operation Control Register

This register is used to determine the clock of 16-bit counter and also used to select motor control PWM operation or 16-bit timer operation.

■ PWM Timer Operation Control Register (PWMTMCNT: 0x03E16)

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	PSCSEL1	PSCSEL0	PWMCK SEL1	PWMCK SEL0	SEL_PWM _TM
At reset	-	-	-	0	0	0	0	0
Access	-	-	-	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-5	-	-
4-3	PSCSEL1 PSCSEL0	Count clock division selection 00: 1/1 clock 01: 1/2 clock 10: 1/4 clock 11: 1/16 clock
2-1	PWMCKSEL1 PWMCKSEL0	Count clock source selection 00: fs 01: Synchronous TMIO input (fs synchronous) 10: fpll-div 11: Synchronous TMIO input (fpll-div synchronous)
0	SEL_PWM_TM	Operation selection 0: PWM operation 1: 16-bit timer operation

Set the SEL_PWM_TM flag to “1” to use 16-bit timer operation. In order to set PWM operation, set the SEL_PWM_TM flag to “0” and select complementary 3 phases PWM output or 4 phases PWM output by output mode selection of RELCTR register.

9.2.13 4 phases PWM Mode Register

This register is used to perform 4 phases PWM operation mode control setting.

■ 4 phases PWM Mode Register (RELCTR: 0x03E17)

bp	7	6	5	4	3	2	1	0
Flag	RELIRQA	RELIRQB	RELIRQC	RELIRQD	-	MD1CTR	RELMD1	RELMD0
At reset	0	0	0	0	-	0	0	0
Access	R/W	R/W	R/W	R/W	-	R/W	R/W	R/W

bp	Flag	Description
7	RELIRQA	First phase interrupt event enable control 0: Disabled 1: Enabled
6	RELIRQB	Second phase interrupt event enable control 0: Disabled 1: Enabled
5	RELIRQC	Third phase interrupt event enable control 0: Disabled 1: Enabled
4	RELIRQD	Forth phase interrupt event enable control 0: Disabled 1: Enabled
3	-	-
2	MD1CTR	PWM output order control in 1-phase mode 0: In inversion setting, inverted after output phase active period 1: In inversion setting, immediately switched to inversion
1-0	RELMD1 RELMD0	PWM output mode selection 00: Complementary 3 phases PWM output 01: 4 phases PWM output (1-phase mode) 10: 4 phases PWM output (2-phase mode) 11: 4 phases PWM output (1-phase to 2-phase mode)

Be sure to set bp0 of the PWMTMCNT to “0” to select PWM output mode.

9.2.14 4 phases PWM Output Order Control Register

This register is used to set 4 phases PWM output order.

- 4 phases PWM Output Order Control Register (PWMODR: 0x03E18)

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	-	-	-	ORDER
At reset	-	-	-	-	-	-	-	0
Access	-	-	-	-	-	-	-	R/W

bp	Flag	Description
7-1	-	-
0	ORDER	4 phases PWM output order 0: Normal rotation 1: Reverse rotation

9.2.15 4 Phases PWM Output Status Register

This register is used to show the active interval in 4 phases PWM mode.

■ 4 Phases PWM Output Status Register (RELSTAT: 0x03E19)

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	-	-	STAT1	STAT0
At reset	-	-	-	-	-	-	0	0
Access	-	-	-	-	-	-	R	R

bp	Flag	Description
7-2	-	-
1-0	STAT1-0	4 phases PWM output status 00: During 1-phase output 01: During 2-phase output 10: During 3-phase output 11: During 4-phase output

9.2.16 PWM Cycle/Compare Collective Setting Register 1

This register is used to collectively set the lower 8 bits of PWM cycle setting register and PWM phase comparison setting register at equivalence. This register is only for writing and the read data is undefined.

■ PWM Cycle/Compare Collective Setting Register 1 (PWMCMP1: 0x03E1A)

bp	7	6	5	4	3	2	1	0
Flag	PWM CMPL7	PWM CMPL6	PWM CMPL5	PWM CMPL4	PWM CMPL3	PWM CMPL2	PWM CMPL1	PWM CMPL0
At reset	0	0	0	0	0	0	0	0
Access	W	W	W	W	W	W	W	W

bp	Flag	Description
7-0	PWMCMPL7 to PWMCMPL0	An equivalence is written to the lower 8-bits of the PWM cycle setting register and PWM phase comparison setting register.

9.2.17 PWM Cycle/Compare Collective Setting Register 2

This register is used to collectively set the upper 8 bits of PWM cycle setting register and PWM phase comparison setting register at equivalence. This register is only for writing and the read data is undefined.

■ PWM Cycle/Compare Collective Setting Register 2 (PWMCMP2: 0x03E1B)

This register is used to collectively set the upper 8 bits of PWM cycle setting register and PWM phase comparison setting register at equivalence. This register is only for writing and the read data is undefined.

bp	7	6	5	4	3	2	1	0
Flag	PWM CMPH7	PWM CMPH6	PWM CMPH5	PWM CMPH4	PWM CMPH3	PWM CMPH2	PWM CMPH1	PWM CMPH0
At reset	0	0	0	0	0	0	0	0
Access	W	W	W	W	W	W	W	W

bp	Flag	Description
7-0	PWMCMPH7 to PWMCMPH0	An equivalence is written to the upper 8-bits of the PWM cycle setting register and PWM phase comparison setting register.

When accessing PWMCMP1 and 2 registers, make sure to access the lower 8 bits (PEMCMP1 register) by word instruction.

9.3 16-bit Timer Operation

9.3.1 Operation

16-bit timer operation is the function which can repeatedly generate interrupts at regular time intervals.

■ 16-bit Timer Operation

The generation cycle of timer interrupts is set by the clock source selection and the set value of PWMSET register, in advance. When binary counter reaches the set value of PWM cycle setting register, an interrupt is generated at the next count clock. Binary counter is cleared at the time of PWMSET compare match. After binary counter is cleared, the counting up is restarted from 0x0000.

Timer 9 can generate another set of an independent interrupt (Timer 9 TCMPA compare match interrupt) by the set value of TCMPA register. At that time, binary counter is cleared as the above setup.

Table:9.3.1 16-bit Timer Interrupt Generation Factor and Binary Counter Clear Factor

PWMMDL, PWMMDH	Interrupt generation factor	Interrupt factor	Binary counter clear factor
INTAEN1	PWMSET compare match	PWMOVIRQ	PWMSET compare match
INTBEN1	Binary counter clear	PWMUDIRQ	
INTCEN1	TCMPA compare match	PWMOC2IRQ	

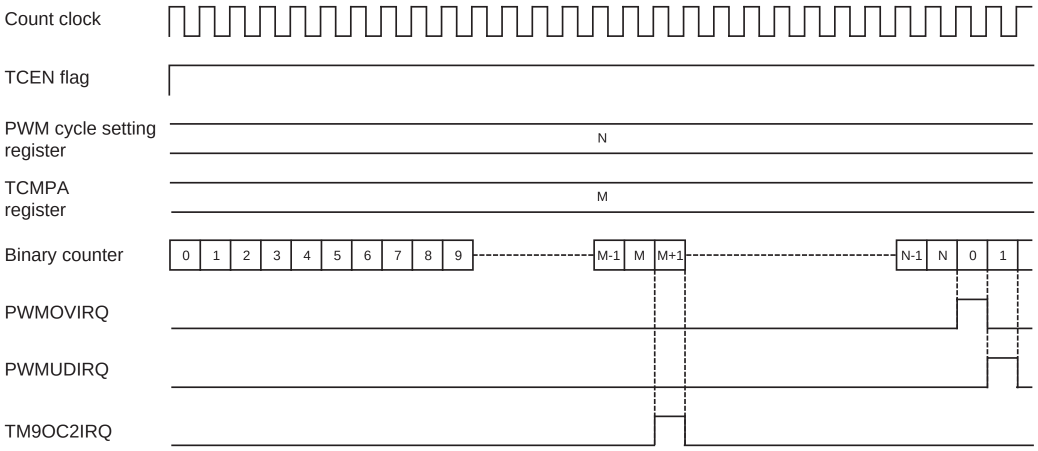


Figure:9.3.1 Interrupt Generation Timing of Motor Control 16-bit Timer

PWM cycle setting register is double buffer type. Thus, when the value of preset registers is changed during the timer counting, the changed value is set to PWM cycle setting register once again at the timing of binary counter is cleared. This function can change the compare value constantly, without disturbing the cycle even during timer operation (Reload function).

When the CPU reads the 16-bit binary counter, the read data is handled in 8-bits units in the LSI even if it is a 16-bit MOVW instruction. As a result, it will read the data incorrectly if a carry from the lower 8 bits to the upper 8 bits occurs during counting operation.

To count properly, do not switch the count clock on the timer operation. To switch the count clock, stop the timer operation.

Under the following conditions, update the value of PWM cycle setting register after setting the PWMCKSEL1 to 0 flags of PWM timer operation control register to “00” (clock source: fs). Then start the timer operation with selecting the clock source to be used.
<Conditions>
The TCEN flag of PWM mode control register = 0 (prohibited),
the SEL_PWM_TM flag of PWM timer operation control register = 1 (16-bit timer operation)
and the PWMCKSEL1 to 0 flags of PWM timer operation control register = 10 (fpll-div)

Clock source can be selected as follows.

Table:9.3.2 Clock Source at Timer Operation

Clock source	1 count time
fpll-div	100 ns
fpll-div/2	200 ns
fpll-div/4	400 ns
fpll-div/16	1.6 µs
fs	200 ns
fs/2	400 ns
fs/4	800 ns
fs/16	3.2 µs

fpll-div = 10 MHz (PLL not used)
fs = fpll-div/2 = 5 MHz

■ Count Timing of Timer Operation (Timer 9)

Binary counter counts up with the selected clock source as a count clock. The basic operation of whole 16-bit timer functions is as below.

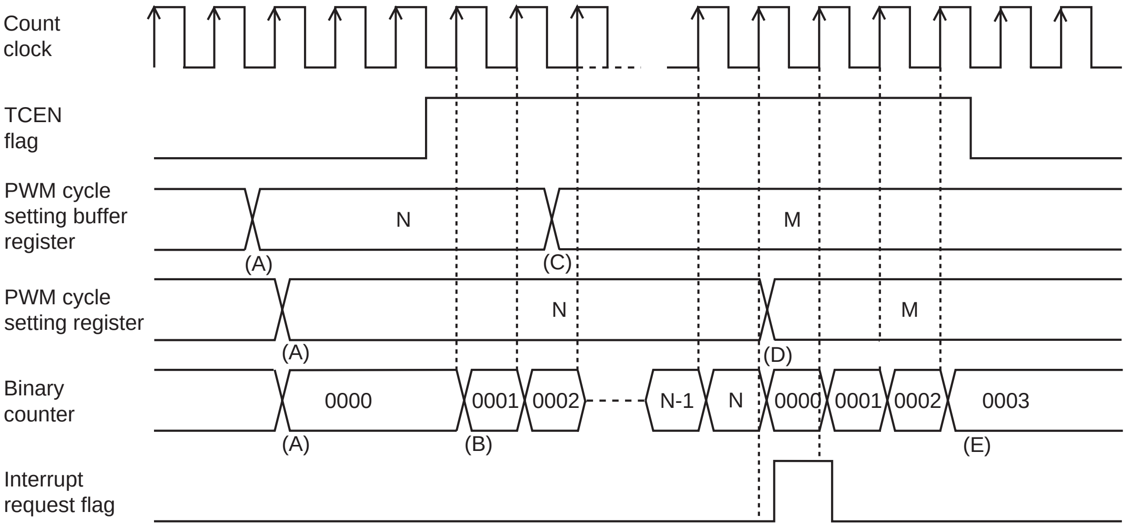


Figure:9.3.2 Count Timing of Timer Operation (Timer 9)

(A) When a value is written to PWM cycle setting buffer register while the TCEN flag is in stop state ("0"), the same value is loaded to PWM cycle setting register at the next count clock of the writing cycle and binary counter is cleared to 0x0000.

(B) When the TCEN flag is in operating state ("1"), binary counter starts counting. The counting is executed at the rising edge of the count clock.

(C) Even if PWM cycle setting buffer register is rewritten when the TCEN flag is in operating state ("1"), binary counter is not changed.

(D) When binary counter reaches the value of PWM cycle setting register, the set value of PWM cycle setting buffer register is loaded to PWM cycle setting register at the next count clock. In addition, the interrupt request flag is set, binary counter is cleared to 0x0000 and counting up starts again.

(E) When the TCEN flag is in stop state ("0"), binary counter is stopped.

When binary counter reaches the value of PWM cycle setting register, the interrupt request flag is set and binary counter is cleared at the next count clock. Thus, set PWM cycle setting register as:
(Set value of PWM cycle setting register) = (Number of counting until an interrupt request is generated - 1)

After a timer interrupt request generation, up to 3 system clocks are needed to generate the next interrupt request flag. Even if compare match occurs during this period, no interrupt request flag is generated.

When Timer 9 TCMPA compare match interrupt is used, change the set value of PWM U-phase compare register (TCMPA) smaller than the set value of PWM cycle setting register (PWMSET).

On the interrupt service routine, clear the timer interrupt request flag before the timer is started.

When the TCEN flag of PWMMD register is changed with other bits at the same time, binary counter may count up by switching operation.

Timer 9 count clock should be set when the timer interrupt is disabled.

When CPU reads binary counter, the read data is handled in 8-bits units in the LSI. As a result, it will read the data incorrectly if a carry from the lower 8 bits to the upper 8 bits occurs during counting operation.
To read the correct value, stop the timer before reading.

Under the following conditions, update the value of PWM cycle setting register after setting the PWMCKSEL1 to 0 flags of PWM timer operation control register to “00” (clock source: fs). Then start the timer operation with selecting the clock source to be used.

<Conditions>
- The frequency of a system clock (fs) is faster than twice the frequency of a high-speed clock for peripheral functions (fpll-div), and
- The following conditions are met.

	Relevant Block	PWMMDL (PWM mode control register)	PWMTMCNT (PWM timer operation control register)	
		TMEN	SEL_PEM_TM	PWMCKSEL1 to 0
Condition	Timer 9	0 (operation stop)	1 (16-bit timer operation)	10 (pll-div)

9.3.2 Setup Example

■ Timer Operation Setup Example

Timer 9 generates an interrupt at regular time intervals to realize the clock function. Select fpll-div/2 (fpll=10 MHz at operation) as the clock source to generate an interrupt every 1000 division (200 µs).
An example setup procedure, with a description of each step is shown below.

Setup Procedure	Description
(1) Stop the counter, select saw to tooth wave PWMMDL(0x03E00) bp1: TCEN =0 bp0: WAVEMD =1	(1) Set the TCEN flag of PWMMD register to “0” to stop Timer 9 counting. In addition, set the WAVEMD flag to “1” to select the saw-tooth wave.
(2) Select the count clock source and timer operation PWMTMCNT(0x03E16) bp4 to 3: PSCSEL1 to 0 =01 bp2 to 1: PWMCKSEL1 to 0 =10 bp0: SEL_PWM_TM =1	(2) Select fpll-div as the clock source by the PWMCKSEL1 to 0 flags of PWMTMCNT register, and select 1/2 of fpll-div as the count clock source by the PSCSEL1 to 0 flags. Also, set the SEL_PWM_TM flag to “1” to select timer operation.
(3) Set the interrupt generation cycle PWMSET(0x03E05, 0x03E04)=0x03E7	(3) Set the value of the interrupt generation cycle in PWMSET register. The cycle is 1000 divisions. As a result, the set value should be 1000-1=999. At the time, Timer 9 binary counter is initialized to 0x0000.
(4) Enable the interrupt PWMMDL(0x03E00) bp4: INTBEN =1 PWMOVICR(0x03FF4) bp1: PWMOVIE =1	(4) Set the INTBEN flag of PWMMDL register to “1” to enable interrupt output. Also, set the PWMOVIE flag of PWMOVICR register to “1” to enable interrupt.
(5) Start the timer operation PWMMDL(0x03E00) bp1: TCEN =1	(5) Set the TCEN flag of PWMMD register to “1” to operate Timer 9.

Binary counter counts up from 0x0000. When binary counter reaches the set value of PWMSET register, Timer 9 interrupt request flag is set at the next count clock and the value of binary counter becomes 0x0000 to start counting up again.

9.4 16-bit Event Count

9.4.1 Operation

Event count operation can be performed with 1/1 (no division), 1/2, 1/4 and 1/16 of the synchronous TM9IO input as the clock source by selection.

■ 16-bit Event Count Operation (Timer 9)

The operation of the event count synchronizes an external signals input into TM9IO pin with fs or fpll-div, and counts the signals by binary counter. If binary counter reaches a preset value of compare register (PWMSET), an interrupt can be generated at the next count clock.

Table:9.4.1 Event Count Input Clock Source

	Timer 9
Event input	Synchronous TM9IOA input
	Synchronous TM9IOB input

If a value of binary counter is read during operation, an incomplete data at the moment of counting up may be read. Also, binary counter may have an unexpected value when the timer is stopped.
To prevent these troubles, use the event count by the synchronous TM9IO input, which is shown in the following page.

■ Count Timing of Synchronous TM9IO Input (Timer 9)

If the synchronous TM9IO input is selected, a synchronous circuit output signal is input to Timer 9 count clock. The synchronous circuit output signal is changed at the rising edge of the system clock (fs or fpll-div) after TM9IO input signal is changed. Binary counter counts up at the falling edge of the synchronous circuit output signal or the synchronous circuit output signal that passed through the division circuit.

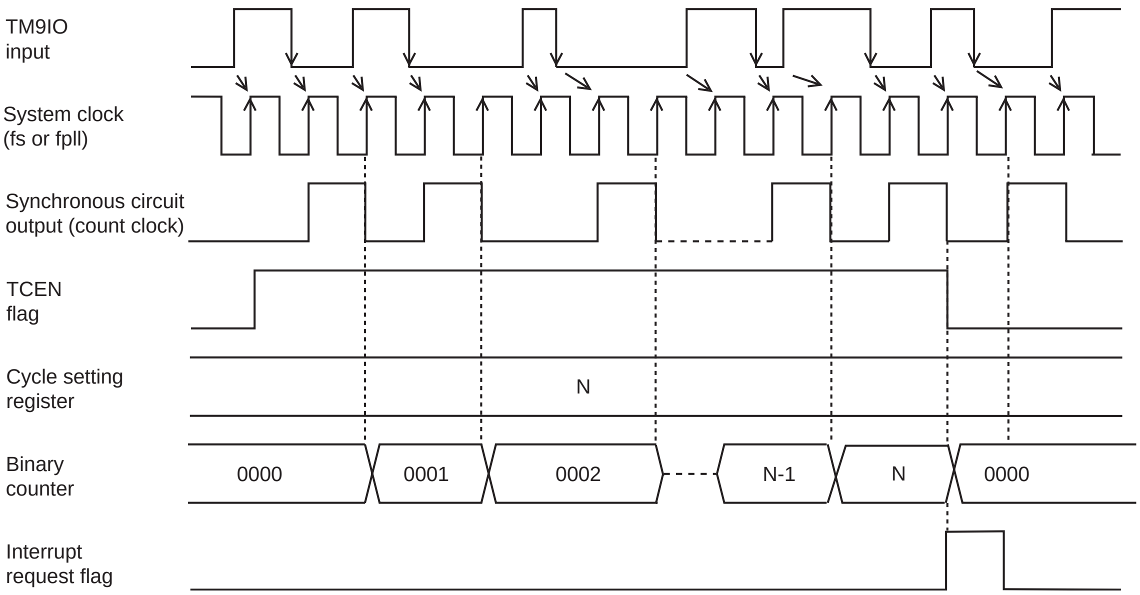


Figure:9.4.1 Count Timing of Synchronous TM9IO Input (Timer 9)

Timer 9 counts up binary counter by a signal in synchronization with the system clock (fs or fpll-div). Thus, collect value is always read when binary counter is read.

Input from TM9IO should be a waveform which has more than 2 times cycle than synchronous clock (fpll-div or fs). If less than the above waveforms are input, it may not be counted correctly.

9.4.2 Setup Example

■ Event Count Setup Example

When the falling edge of TM9IOA input pin signal is detected 5 times with using Timer 9, an interrupt is generated.

An example setup procedure, with a description of each step is shown below.

Setup Procedure	Description
(1) Stop the counter, select saw to tooth wave PWMMDL (0x03E00) bp1: TCEN =0 bp0: WAVEMD =1	(1) Set the TCEN flag of PWMMD register to “0” to stop Timer 9 counting. In addition, set the WAVEMD flag to “1” to select the saw-tooth wave.
(2) Select the timer input TMINSEL2(0x03FB3) bp7 to 6: TMINSEL27 to 26 =00	(2) Set the TMINSEL27 to 26 flags of TMINSEL2 register to “00” to set PA7 pin as timer input.
(3) Set the special function pin to input PADIR(0x03E9A) bp7: PADIR7 =0	(3) Set the PADIR7 flag of PADIR register to “0” to set PA7 pin to input mode. Add a pull-up resistor, if necessary. [Chapter 5 I/O Port]
(4) Select the count clock source PWMTMCNT(0x03E16) bp4 to 3: PSCSEL1 to 0 =00 bp2 to 1: PWMCKSEL1 to 0 =11 bp0: SEL_PWM_TM =1	(4) Select synchronous TMIO input (fpll-div synchronous) as the clock source by the PWMCKSEL1 to 0 flags of PWMTMCNT register. In addition, select 1/1 as the count clock source by the PSCSEL1 to 0 flags. Set the SEL_PWM_TM flag to “1” to select timer operation.
(5) Set the interrupt generation cycle PWMSET(0x03E05, 0x03E04)=0x0004	(5) Set the value of the interrupt generation cycle in PWMSET register. The set value should be 4 to count 5 times. At that time, Timer 9 binary counter is initialized to 0x0000.
(6) Enable the interrupt PWMMDL(0x03E00) bp4: INTBEN =1 PWMOVICR(0x03FF4) bp1: PWMOVIE =1	(6) Set the INTBEN flag of PWMMDL register to “1” to enable interrupt output. Also, set the PWMOVIE flag of PWMOVICR register to “1” to enable interrupt.
(7) Start the event count PWMMDL(0x03E00) bp1: TCEN =1	(7) Set the TCEN flag of PWMMDL register to “1” to start Timer 9 operation.

Binary counter counts up from 0x0000 every time a falling edge of TM9IO input is detected. When binary counter reaches the set value of PWMSET register, Timer 9 interrupt request flag is set at the next count clock and the value of binary counter becomes 0x0000 to start counting up again.

If the above procedures (4) to (7) are not followed, it can cause improper operation.

9.5 16-bit Timer Pulse Output

9.5.1 Operation

Timer pulse output function can output pulse signal with an arbitrary frequency from TM9IO pin. Also, PWM output operation can output an arbitrary pulse width periodically from TM9OD pin. Switching between 16-bit timer operation and PWM operation can be set by the SEL_PWM_TM flag of PWM timer operation control register.

Pulse output in 16-bit timer operation setting and PWM output in PWM operation setting are shown below.

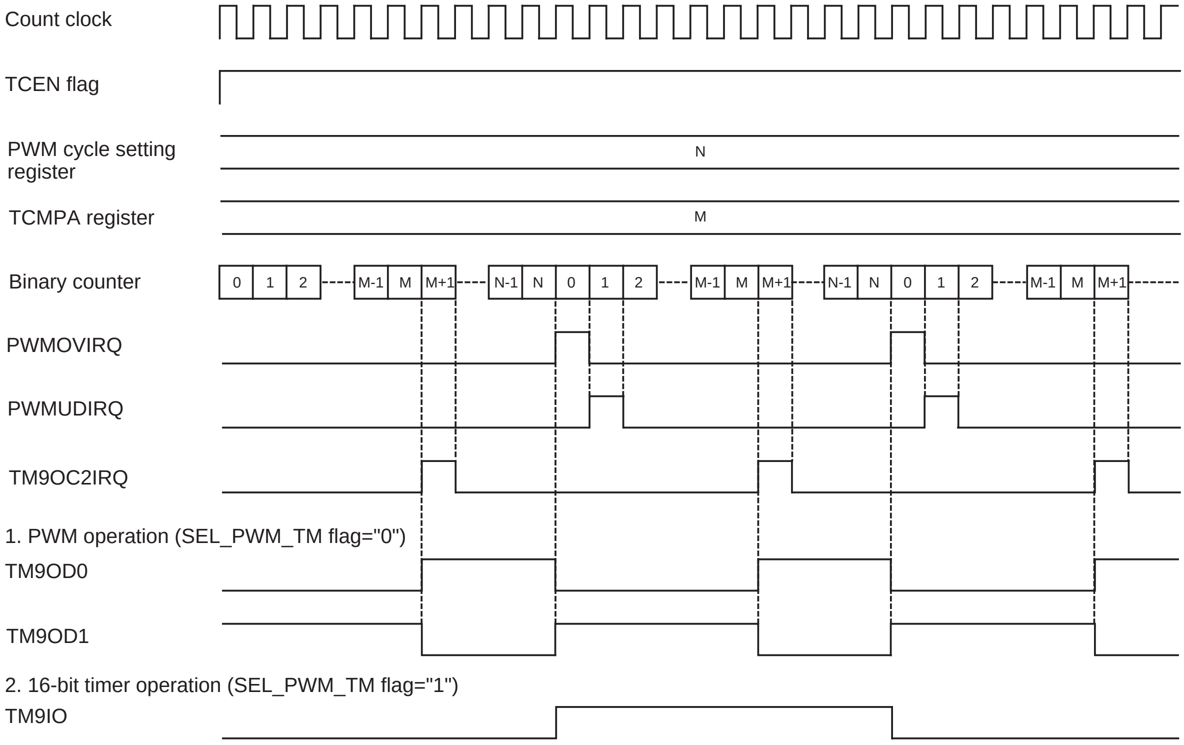

Count clock
TCEN flag
PWM cycle setting register
N
TCMPA register
M
Binary counter
0 1 2 M-1 M M+1 N-1 N 0 1 2 M-1 M M+1 N-1 N 0 1 2 M-1 M M+1
PWMOVIRQ
PWMUDIRQ
TM9OC2IRQ
1. PWM operation (SEL_PWM_TM flag="0")
TM9OD0
TM9OD1
2. 16-bit timer operation (SEL_PWM_TM flag="1")
TM9IO

Figure:9.5.1 Timing Chart of 16-bit Timer Pulse Output

9.5.2 Setup Example

■ Timer Pulse Output Setup Example

TM9IOA output pin outputs a 25 kHz pulse with using timer 9. Select fpll-div as the clock source and set 1/2 cycle (25 kHz) to PWM cycle setting register to output 25kHz pulse (at fpll-div=10 MHz).

An example setup procedure, with a description of each step is shown below.

Setup Procedure	Description
(1) Stop the counter, select saw to tooth wave PWMMDL (0x03E00) bp1: TCEN =0 bp0: WAVEMD =1	(1) Set the TCEN flag of PWMMD register to “0” to stop Timer 9 counting. In addition, set the WAVEMD flag to “1” to select the saw-tooth wave.
(2) Set the special function pin P8OMD(0x03EBA) bp7: PAOMD7 =1 PADIR(0x03E9A) bp7: PADIR7 =1	(2) Set the PAOMD7 flag of A registers to “1” to set PA7 pin as a special function pin. In addition, set the PADIR7 flag of PADIR register to “1” to set PA7 pin to output mode. [Chapter 5 I/O Port]
(3) Select the count clock source PWMTMCNT(0x03E16) bp4 to 3: PSCSEL1 to 0 =00 bp2 to 1: PWMCKSEL1 to 0 =11 bp0: SEL_PWM_TM =1	(3) Select synchronous TMIO input (fpll-div synchronous) as the clock source by the PWMCKSEL1 to 0 flags of PWMTMCNT register. Besides, select 1/1 as the count clock source by the PSCSEL1 to 0 flags. Set the SEL_PWM_TM flag to “1” to select timer operation.
(4) Set the timer pulse output generation cycle PWMSET(0x03E05, 0x03E04) =0x00C7	(4) Set 1/2 of the timer pulse output cycle to PWMSET register. To set 50 kHz by dividing 10 MHz, set as; 200-1=199 (0xC7) At the same time, TM9BC is initialized to 0x0000.
(5) Start the timer operation PWMMDL(0x03E00) bp1: TCEN =1	(5) Set the TCEN flag of PWMMDL register to “1” to start Timer 9 operation.

Binary counter counts up from 0x0000. If binary counter reaches the set value of PWMSET register and is cleared to 0x0000, the signal of TM9IO output is inverted and binary counter counts up from 0x0000 again.

9.6 Complementary 3 phases PWM

9.6.1 Operation

■ Complementary 3 phases PWM Waveform Mode

Waveform mode can be set by the WAVEMD flag of PWMMD register. When the flag is set to "0", triangular waves are specified. When the flag is set to "1", saw-tooth waves are specified.

Table:9.6.1 shows the output waveform logical operation expression and output level.

Table:9.6.1 Logical Operation Expression and Output Level

Logical operation expression	TM9OD0, 2, 4	TM9OD1, 3, 5
Value to be compared ≤ Counter value	High	Low
Value to be compared > Counter value	Low	High

When triangular waves are set, the counter counts up and down. It counts the cycle setting value twice at the end of counting up, and counts "0" twice at the end of counting down. When saw-tooth waves are set, the counter counts up. After the counter counts up until the cycle setting value, it becomes "0" at the next counting. Triangular and saw-tooth wave output figures are shown below.

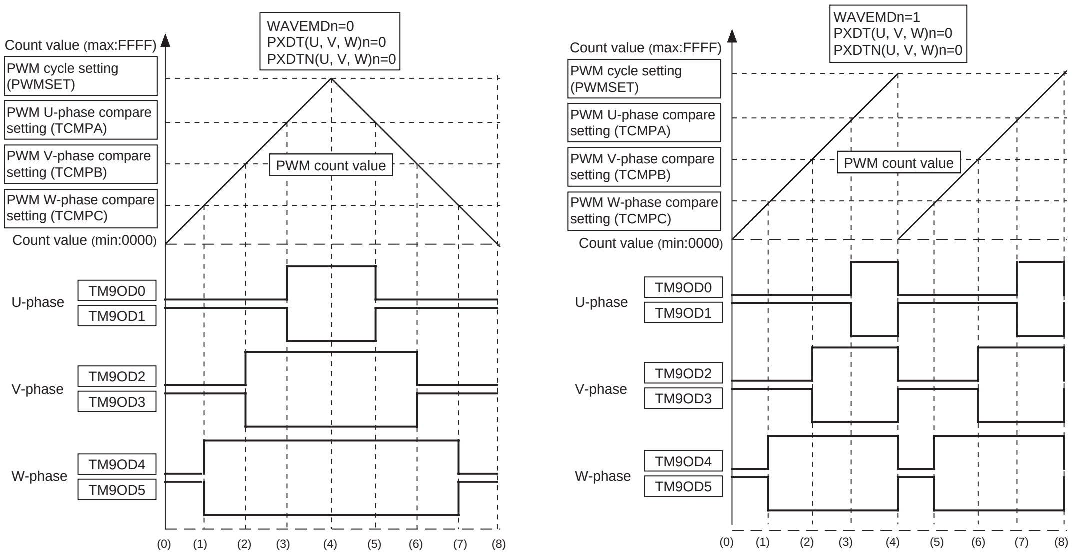


Figure:9.6.1 Triangular and Saw-tooth Waves

Set TCMPA, TCMPB, and TCMPC to meet the following condition.

TCMPA/TCMPB/TCMPC ≤ PWMSET

■ Setting PWM Period

The 3 phases cycle for PWM is set by PWMSET register. PWM counting is operated by PWMBC. The expression of the PWM cycle is as follows. The count clock of PWMBC is set by the PWMTMCNT flag.

Waveform mode	PWM period
Triangular wave	Count clock period × (PWMSET set value +1) × 2
Saw-tooth wave	Count clock period × (PWMSET set value +1)

■ Starting and Stopping PWM Output

PWM output starts when the TCEN flag of PWMMD register is set to "1", and stops when the flag is set to "0". Table:9.6.2 shows Timer 9 status when stopping.

Table:9.6.2 PWM Status When Counting Operation is Disabled

PWM block		Status
Phase output	When the output polarity positive phase is selected	Low
	When the output polarity negative phase is selected	High
PWM binary counter (16-bit counter)		PWM operation: 0, timer operation: retained
Dead Time counter		Reset status
PWM control register (double-buffer)		Double-buffer data loaded
PWM control register (single-buffer)		Retained

When the timer operation is selected, binary counter retains the count value without clearing. To clear binary counter during timer operation selection, set the TCEN to "0" once, and then access the cycle setting register.

■ Output Waveform Polarity

OUTMD register can be used to control the polarity of the PWM output waveform.

Table:9.6.3 shows the relationship between flags and set values. When the PXDT flag is set to "1", PWM output is switched to NPWM output.

Table:9.6.3 Flags and Settings

	Flag	Setting value	
		0	1
TM9OD0	PXDTU	Positive phase	Negative phase
TM9OD1	PXDTNU	Negative phase	Positive phase
TM9OD2	PXDTV	Positive phase	Negative phase
TM9OD3	PXDTNV	Negative phase	Positive phase
TM9OD4	PXDTW	Positive phase	Negative phase
TM9OD5	PXDTNW	Negative phase	Positive phase

Figure:9.6.2 shows the flag value and the output waveform.

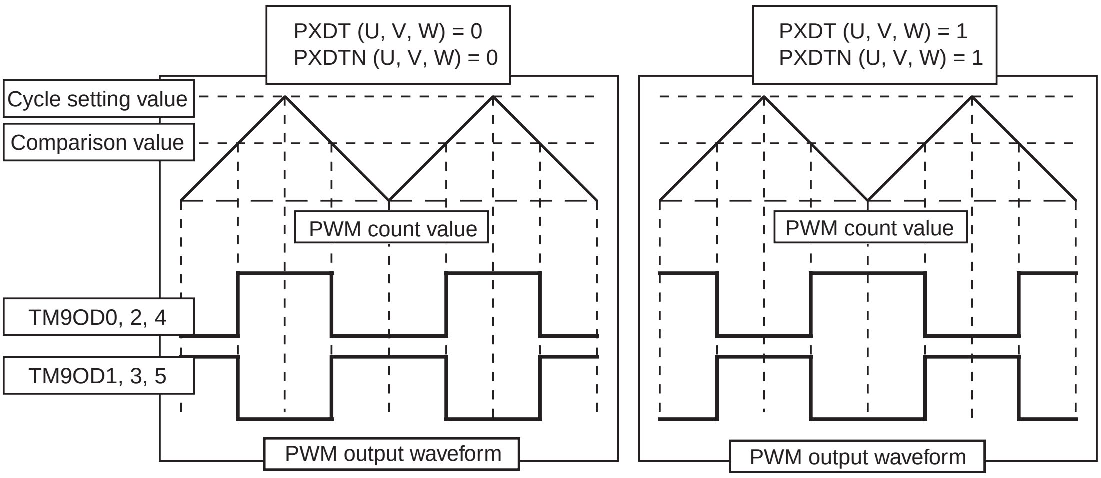


Figure:9.6.2 Flag Value and Output Waveform

■ Double Buffer

Each of Timer 9 is double-buffered to allow data changes during operation. Registers read from and written to by the microcontroller are independent of registers referenced by Timer 9. This makes it possible for microcontroller's register values to be loaded into PWM's registers in synchronization with Timer 9 cycle. However, PWMMD register is single-buffered as it is a basic register that controls timer 9 operation mode. In addition, PWMOFF, IRQCULL, PWMTMCNT, RELCTR and PWMODR registers are also single-buffered. The other PWM control registers are double-buffered or switchable between double-buffered and single-buffered. Therefore, the configuration that best suits the application can be selected. Check the PWM control register list for buffer configuration. Double buffer load timing can be set by overflow and underflow of PWM cycle. PWMMD register can be used to enable or disable either of these timings.
In addition, if binary counter is stopped with counting operation disabled, the double buffer values are directly loaded into timer 9 registers.

Table:9.6.4 Buffer Configuration Available with PWM Control Registers

Register	Double-buffered	Single-buffered	Remarks
PWMMD	-	√	-
OUTMD	√	√	Switching by the SDSLAn flag in PWMMD register
PWMSEL	√	√	Switching by the SDSLBn flag in PWMMD register
PWMSET	√	-	-
TCMPA	√	-	-
TCMPB	√	-	-
TCMPC	√	-	-
DTMSET	√	-	-
DTMSET1	√	-	-
PWMOFF	-	√	-
IRQCULL	-	√	-
PWMTMCNT	-	√	-
RELCTR	-	√	-
PWMODR	-	√	-
PWMCMP1	√	-	-
PWMCMP2	√	-	-

The values of single-buffered registers are reflected immediately.

■ Double Buffer Load Timing

Double buffer load can be enabled by the PCRAEN flag and PCRBEN flag of PWMMD register. Table:9.6.5 shows the relationship between the double buffer load timing and enable setting flag.

Table:9.6.5 Double Buffer Load Timing and Enable Setting Flag

Load timing	Flag (Register)	Description	
At underflow	PCRAEN (PWMMD)	0	Disabled
		1	Enabled
At overflow	PCRBEN (PWMMD)	0	Disabled
		1	Enabled

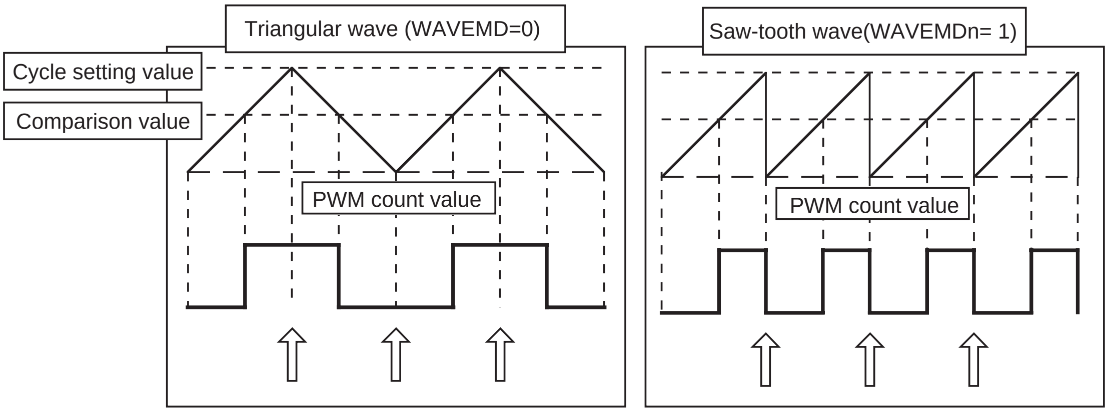


Figure:9.6.3 Double Buffer Load TIming

■ Setting Interrupt Timing

Interrupt signal can be generated in synchronization with Timer 9 cycle. Interrupt signal is generated when an underflow or overflow is generated or when PWM U-phase comparison setting register reaches the set value of binary counter. Table:9.6.6 shows the relationship between the interrupt timing and the enable setting flags.

Table:9.6.6 Interrupt Timing and Enable Setting Flag

Load timing	Flag (Register)	Description		Interrupt source
At underflow	INTAEN(PWMMD)	0	Disabled	PWMUDIRQ
		1	Enabled	
At overflow	INTBEN(PWMMD)	0	Disabled	PWMOVIRQ
		1	Enabled	
TCMPA register reaches the value of binary counter	INTCEN(PWMMD)	0	Disabled	TM9OC2IRQ
		1	Enabled	

■ Dead Time

Dead Time is designed to insert ON time delay into each of the upper and lower phases when the signal is inverted at each phase of PWM output. The DTEN flag of PWMMD register is used to select whether to enable or disable dead time. The ORMD flag of PWMMD register is used to select output logic at dead time insertion. DTMSET register is used to specify delay time inserted as dead time.
Any of "00" to "FF" can be selected as dead time with 8-bit data. The dead time counter functions in synchronization with the clock source set by the PWMCKSEL1 to 0 flags of PWMTMCNT register, and counts by 1 every 2 clock cycles. Calculate the dead time or delay time based on "set value × 2+1".
Thus, when "00" is specified, 1 clock cycle of dead time is inserted if dead time is enabled.

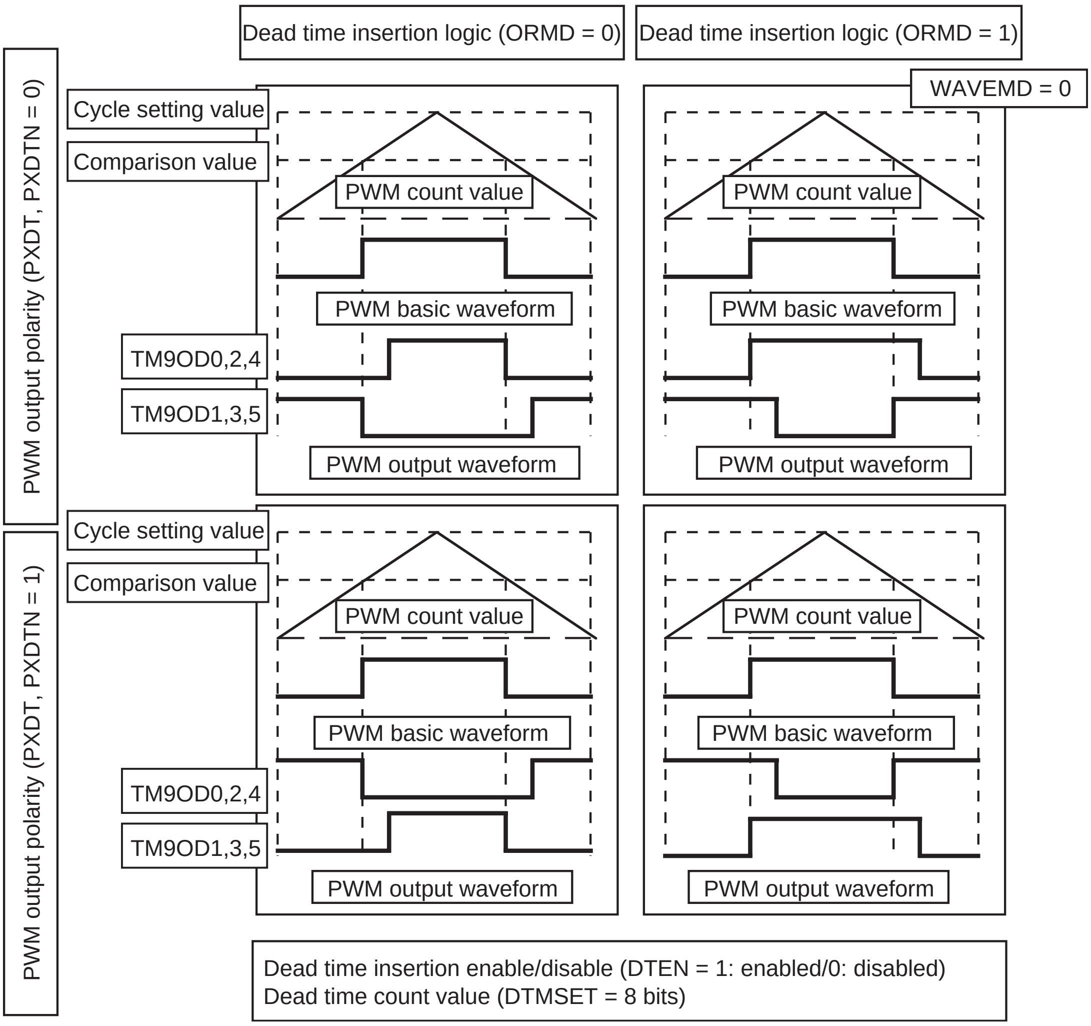

Dead time insertion logic (ORMD = 0)
Dead time insertion logic (ORMD = 1)
WAVEMD = 0
PWM output polarity (PXDT, PXDTN = 0)
Cycle setting value
Comparison value
PWM count value
PWM basic waveform
TM9OD0,2,4
TM9OD1,3,5
PWM output waveform
PWM output polarity (PXDT, PXDTN = 1)
Cycle setting value
Comparison value
PWM count value
PWM basic waveform
TM9OD0,2,4
TM9OD1,3,5
PWM output waveform
Dead time insertion enable/disable (DTEN = 1: enabled/0: disabled)
Dead time count value (DTMSET = 8 bits)

Figure:9.6.4 Dead Time

Though only a single-phase of PWM basic waveform is shown in Figure:9.6.4, PWM 3 phases can be output respectively.

■ Dead Time Insertion Mode Setting

Different dead times can be set in TM9OD0, 2, 4, and TM9OD1, 3, 5 of the PWM output by the DTSEL flag of PWMMD register.
Delay time inserted in dead time insertion mode is set by DTMSET register for TM9OD0, 2, 4. It is also set by DTMSET1 register for TM9OD1, 3, 5.

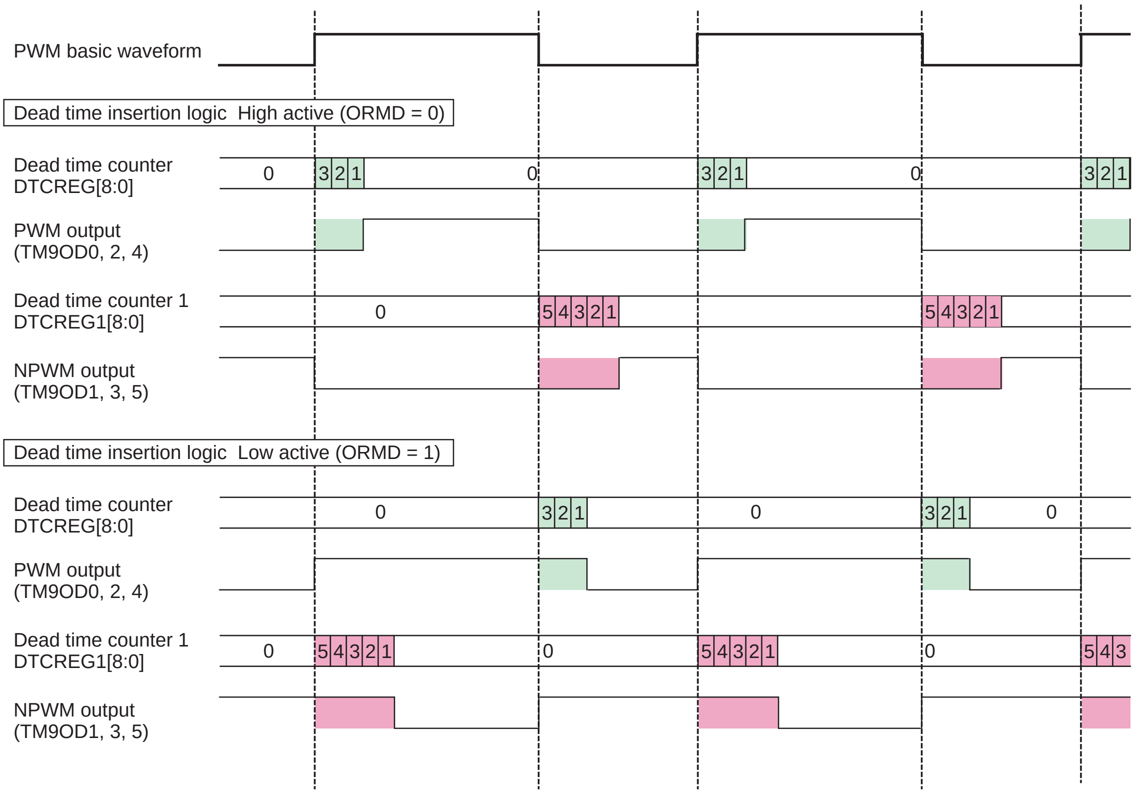

PWM basic waveform
Dead time insertion logic High active (ORMD = 0)
Dead time counter DTCREG[8:0]
0
3 2 1
0
3 2 1
0
3 2 1
PWM output (TM9OD0, 2, 4)
Dead time counter 1 DTCREG1[8:0]
0
5 4 3 2 1
5 4 3 2 1
NPWM output (TM9OD1, 3, 5)
Dead time insertion logic Low active (ORMD = 1)
Dead time counter DTCREG[8:0]
0
3 2 1
0
3 2 1
0
PWM output (TM9OD0, 2, 4)
Dead time counter 1 DTCREG1[8:0]
0
5 4 3 2 1
0
5 4 3 2 1
0
5 4 3
NPWM output (TM9OD1, 3, 5)

Figure:9.6.5 Dead Time Insertion Mode Setting

Though only a single-phase of PWM basic waveform is shown in Figure:9.6.5, PWM 3 phases can be output respectively.

■ High/Low Level Output Setting

PWM output or High/Low level output can be selected for each of six PWM pins by the PSELN02 to 00 flags and the PSEL02 to 00 flags of PWMSEL register. When High/Low level output is selected, Low level output or High level output can be selected by the OTLVN02 to 00 and OTLV02 to 00.
The output timing of High/Low level output is shown below.

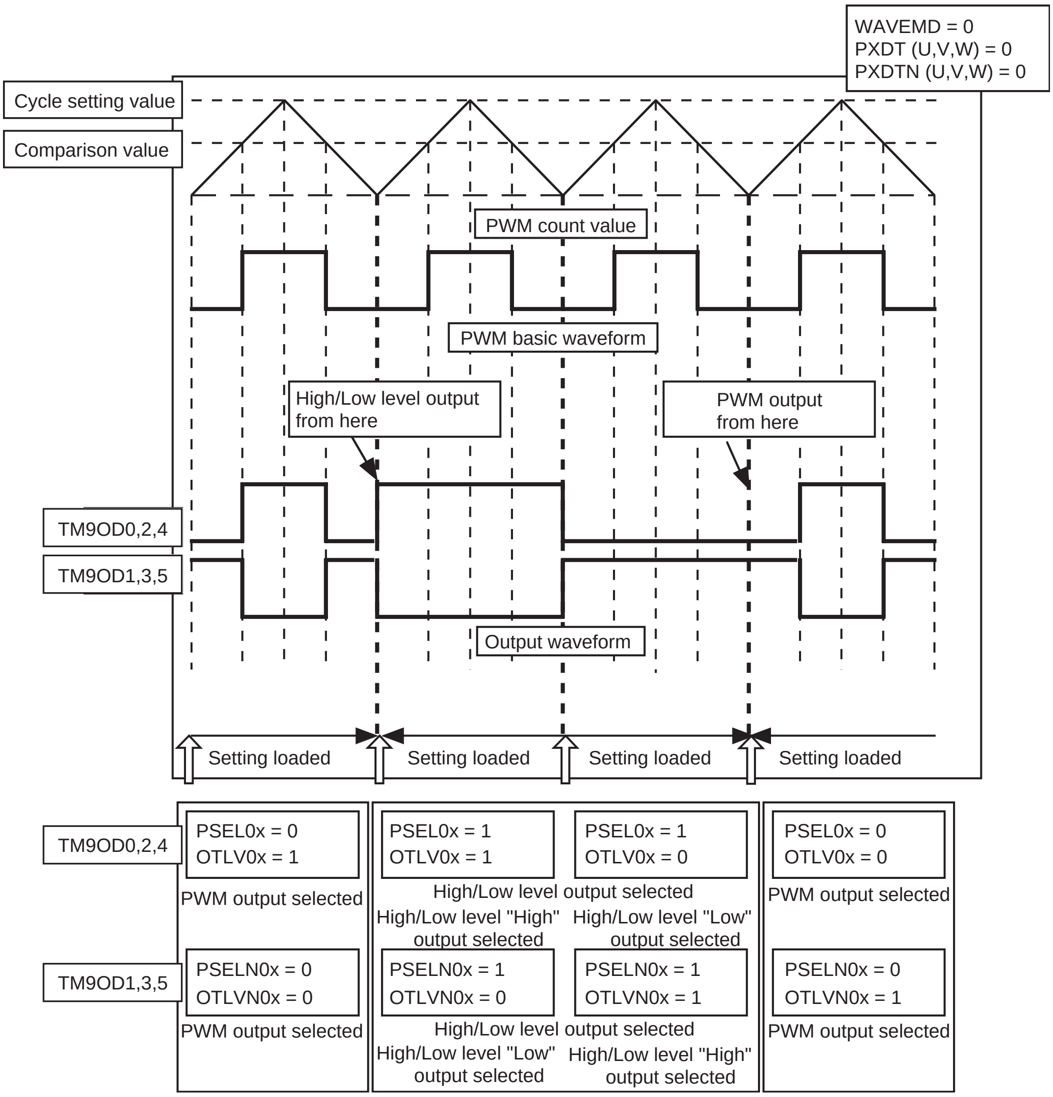


Figure:9.6.6 High/Low Level Output Timing

Though only a single-phase of PWM basic waveform is shown in Figure:9.6.6, PWM 3 phases can be output respectively.

■ Dead Time Insertion at High/Low Level Output

Dead time is inserted as delay time when signals are switched. As a result, dead time is also inserted when PWM output is switched over to High/Low level output. The timing of dead time insertion is shown below.

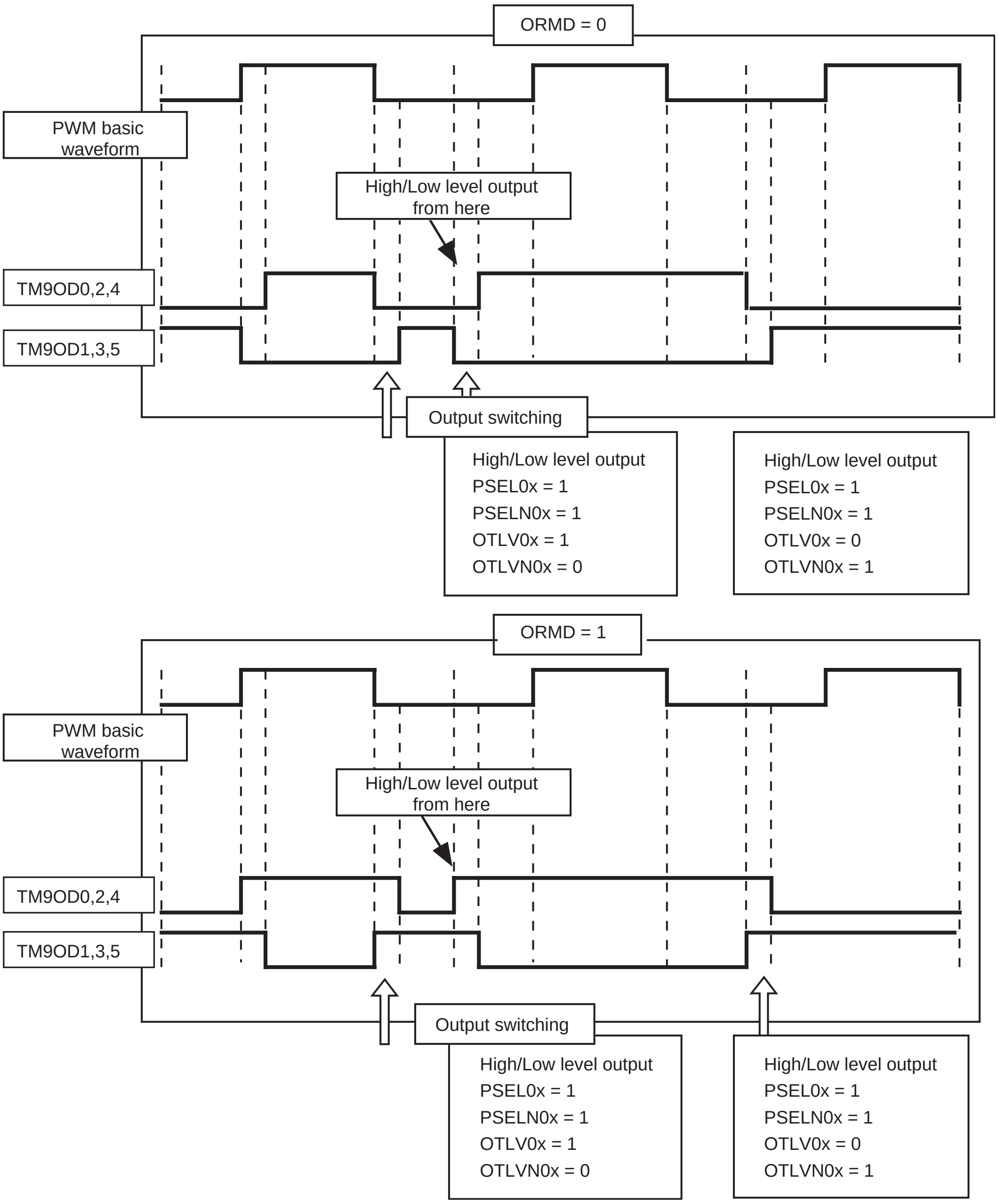


Figure:9.6.7 Dead Time Insertion Timing at High/Low Level Output

Though only a single-phase of PWM basic waveform is shown in Figure:9.6.7, PWM 3 phases can be output respectively.

■ PWM Output Protection Function

Each PWM can change the pin output forcibly by using external interrupt as a trigger.
The changeable pin outputs can be selected from Hi-Z output and inactive output, and all outputs of the six phases can be set individually.
Inactive output follows ORMD (dead time insertion logic) of PWMMD register. In order to cancel the output protection function, set not to use the output protection.
Setup example is shown below.

Table:9.6.7 Setup Example of PWM Output Protection

Flag Name	Description
ORMD = 0	Inactive output is Low output
IRQSEL = 0x1	External interrupt IRQ0 is selected
PRTAU = 0x1	Hi-Z output
PRTANU = 0x2	Inactive output
PRTAV = 0x0	Unused
PRTANV = 0x1	Hi-Z output
PRTAW = 0x2	Inactive output
PRTANW = 0x0	Unused
OUTEN0 = 1	Hi-Z output enabled

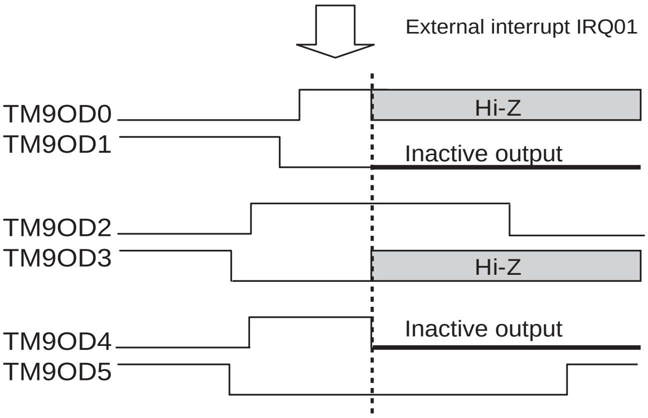


Figure:9.6.8 PWM Output Protection Function

After the external interrupt is input as the trigger, maximum 2 cycles is required until the output is changed by the PWM output protection function.

■ Interrupt Factor Control Function

Interrupt factor control function can control interrupt factors, which is output when binary counter underflows, overflows, or reaches the value of PWM U-phase comparison setting register, to output once every two times or once every four times. The relationship between interrupt factor timing and setting value are shown below.

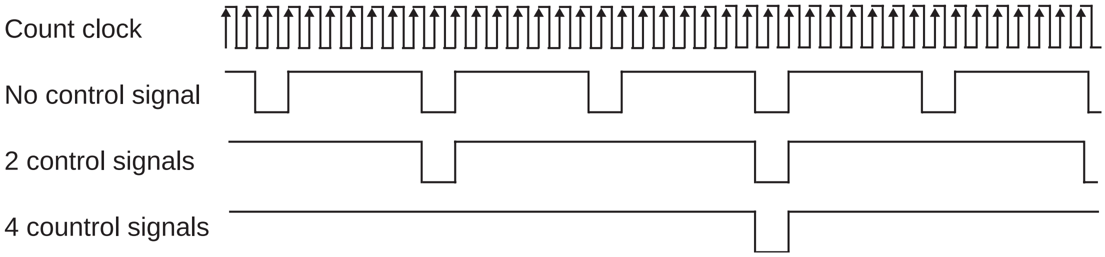


Figure:9.6.9 Relationship between Interrupt Factor Timing and Setting Value

9.7 4 phases PWM

9.7.1 Operation

4 phases PWM can generate multiple phases of PWM waveform. This function can realize 4 phases PWM waveform for ultrasonic monitor driving, stepping motor 1-phase excitation, 2-phase excitation and 1- to 2- phases excitation PWM waveform output.

■ Basic Waveform of 4 phases PWM

Basic waveform generated by 4 phases PWM is shown below.

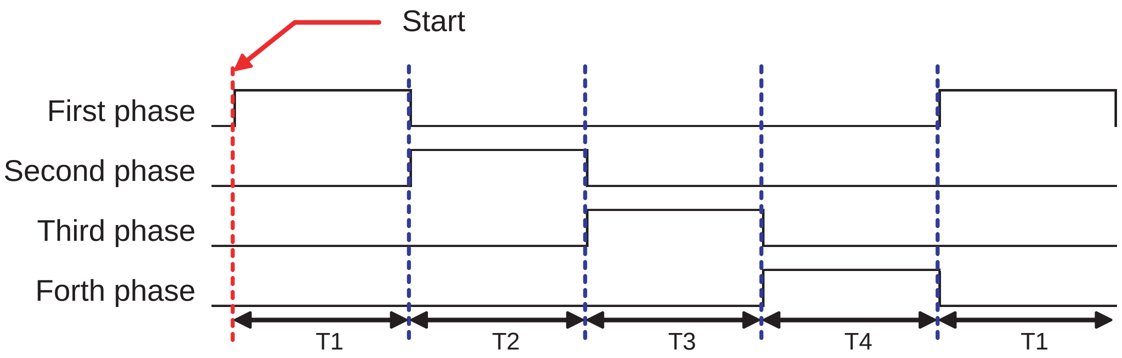


Figure:9.7.1 Basic Waveform 1 (1-phase Mode, RELCTR.md1ctr = 1)

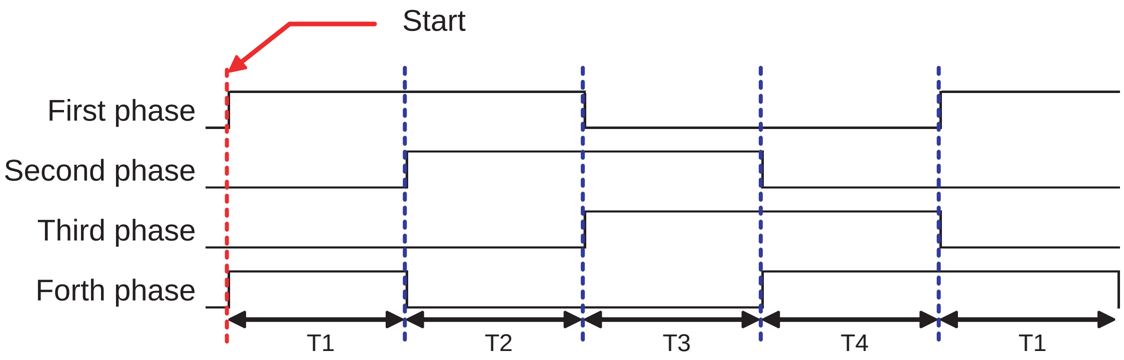


Figure:9.7.2 Basic Waveform 2 (2-phase Mode)

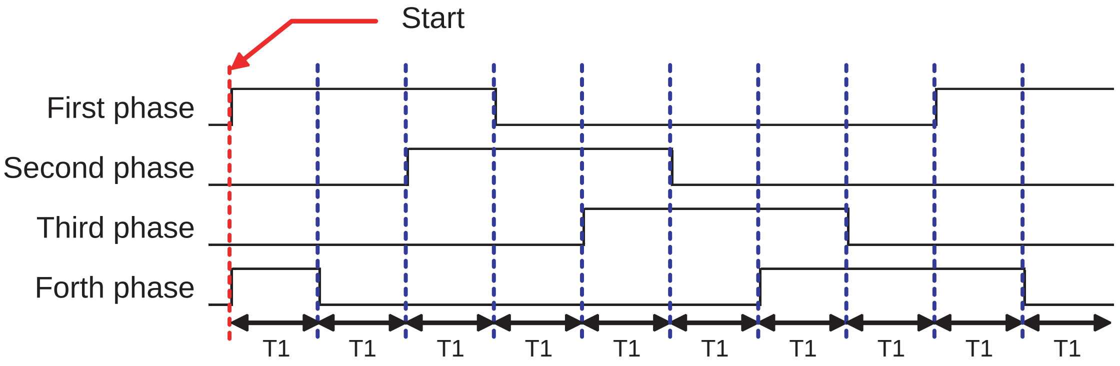


Figure:9.7.3 Basic Waveform 3 (1- to 2-phases Mode)

Each interval of T1, T2, T3 and T4 shown in Figure:9.7.1 to 9.7.3 is determined by registers.
Also, the registers can invert output order of PWM waveform.
The first mode's waveform is shown below.

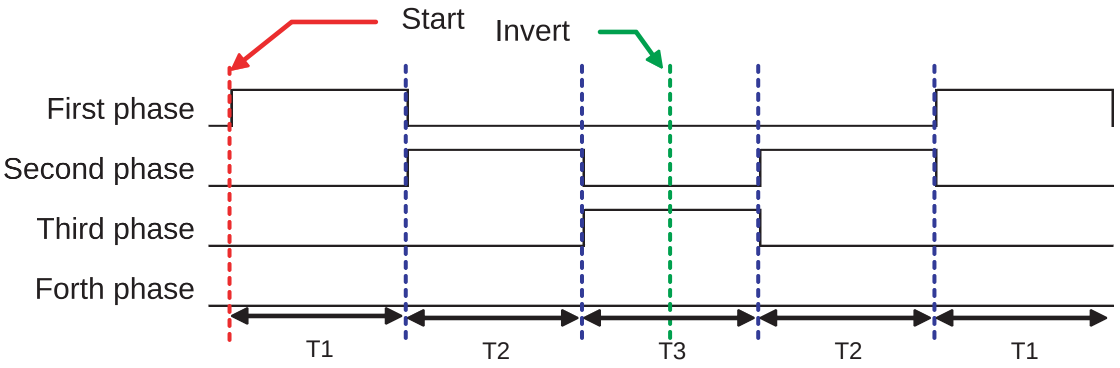


Figure:9.7.4 Basic Waveform 1 Dynamic Inversion (First Mode, RELCTR.md1ctr = 0)

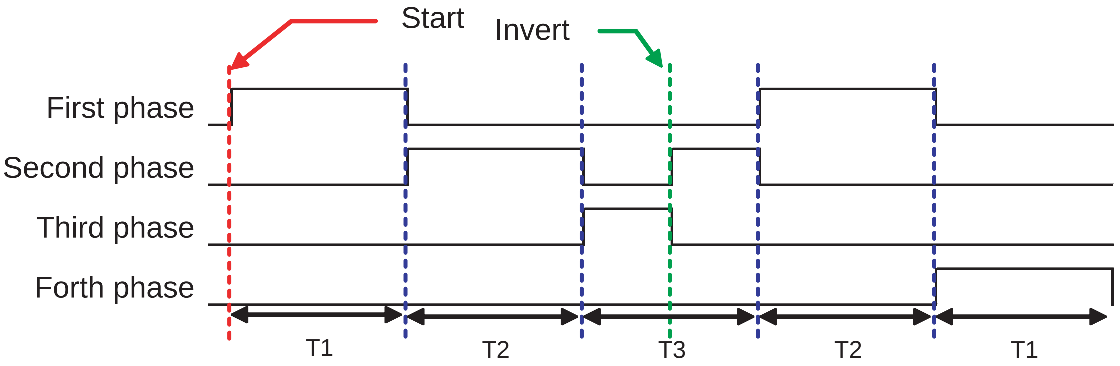


Figure:9.7.5 Basic Waveform 1 Dynamic Inversion (First Mode, RELCTR.md1ctr = 1)

■ Dead Time Insertion

Dead time inserts delay to the starting point of active intervals of each phase.
The waveform when the dead time is inserted is shown below.

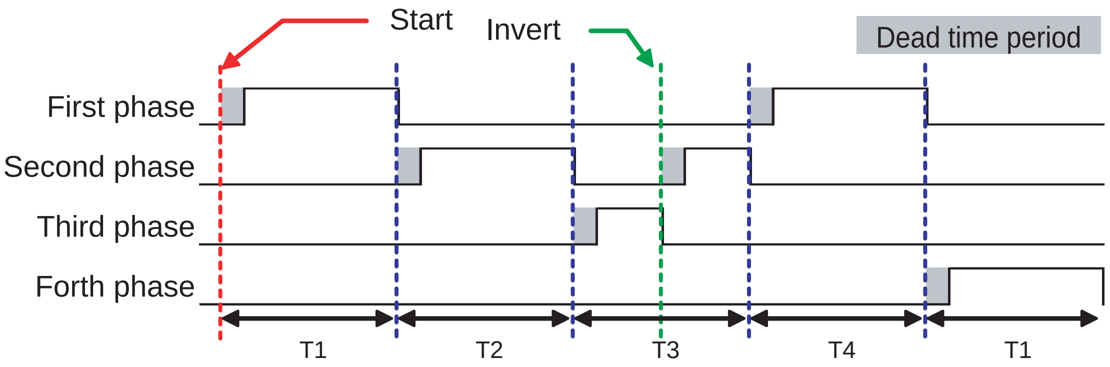


Figure:9.7.6 Dead Time Insertion Basic Waveform 1 (1-phase Mode)

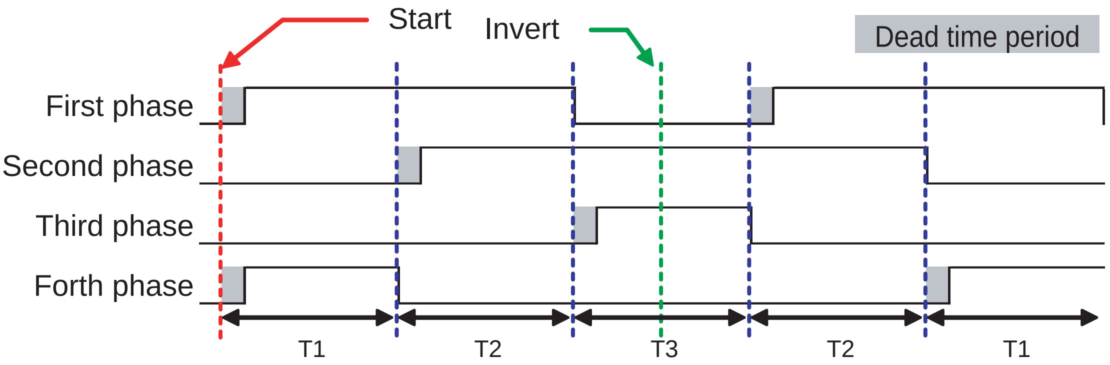


Figure:9.7.7 Dead Time Insertion Basic Waveform 2 (2-phase Mode)

PWM waveform from the first phase to the forth phase are assigned to TM9OD0, 2, 4 and 1.

9.7.2 Setup Example

An example setup procedure, with a description of 4 phases PWM 2-phase mode is shown below.

Setup Procedure	Description
(1) Stop the counter, select saw to tooth wave PWMMDL(0x03E00) bp1: TCEN = 0 bp6: PCRBEN = 1 bp0: WAVEMD = 1	(1) Set the TCEN flag of PWMMD register to "0" to stop Timer 9 counting. In addition, set the PCRBEN flag to “1” to set load timing enable set the WAVEMD flag to "1" to select the saw-tooth wave.
(2) Set the special function pin output P8OMD(0x03EB8) bp0: P8OMD0 = 1 bp1: P8OMD1 = 1 bp2: P8OMD2 = 1 bp3: P8OMD3 = 1 P8DIR(0x03E98) bp0: P8DIR0 = 1 bp1: P8DIR1 = 1 bp2: P8DIR2 = 1 bp3: P8DIR3 = 1	(2) Set the P8OMD0 to 3 flags of P8OMD register to “1” and to select TM90D0 to 3. Besides, Set the P8DIR0 to 3 flags of P8DIR register to "1" to set the output mode. [Chapter 5 I/O Port]
(3) Set the 2 to phase PWM output PWMTMCNT(0x03E16) bp0: SEL_PWM_TM = 0 RELCTR(0x03E17) bp1 to 0: RELMD = 10	(3) Set the SEL_WOM_TM flag of PWM timer operation control register to “0” to select the PWM operation. In addition, set the RELMD flag of 4 phases PWM mode register to “10” to select the 4 phases PWM 2-phase mode.
(4) Set the PWM output cycle PWMCMP(0x3E1B, 0x3E1A) = 0x0031	(4) Set the value of the PWM output cycle in PWMCMP register. To set 50 kHz by dividing 10 MHz, set as; (200/4) - 1=49(0x0031). At the same time, TM9BC is initialized to 0x0000.
(5) Set the PWM dead time DTMSET(0x3E0E, 0x3E0D) = 0x0000 PWMMDL(0x03E00) bp3: DTEN = 1	(5) Set the dead time value of the PWM output phase change in dead time setting register (setting value n - 1). In addition, set the DTEN of PWM mode control register to “1” to enable dead time insertion.
(6) Start the timer operation PWMMDL(0x03E00) bp1: TCEN = 0	(6) Set the TCEN flag of PWM mode control register to “1” to start Timer 9 operation.

Chapter 10 Time Base Timer / Free-run Timer

10.1 Overview

This LSI has a time base timer and a 8-bit free-run timer (Timer 6).

Time base timer is a 15-bit timer counter.

10.1.1 Functions

Table:10.1.1 shows the clock source and the interrupt generation cycle that timer 6 and time base timer can use.

Table:10.1.1 Clock Source and Generation Cycle

	Time base timer	Timer 6 (8-bit free-run)
8-bit timer operation	-	√
Interrupt source	TBIRQ	TM6IRQ
Clock source	fpll-div	fpll-div
	fx	fs
	-	fx
		fpll-div $\times 1/2^{12}$ *1
		fpll-div $\times 1/2^{13}$ *1
		fx $\times 1/2^{12}$ *1
		fx $\times 1/2^{13}$ *1
		synchronous fpll-div $\times 1/2^{12}$ *1
		synchronous fpll-div $\times 1/2^{13}$ *1
		synchronous fx $\times 1/2^{12}$ *2
		synchronous fx $\times 1/2^{13}$ *2
Interrupt generation cycle	fpll-div $\times 1/2^{7}$	The interrupt generation cycle is decided by the arbitrary value written to TM6OC.
	fpll-div $\times 1/2^{8}$	
	fpll-div $\times 1/2^{9}$	
	fpll-div $\times 1/2^{10}$	
	fpll-div $\times 1/2^{13}$	
	fpll-div $\times 1/2^{15}$	
	fx $\times 1/2^{7}$	
	fx $\times 1/2^{8}$	
	fx $\times 1/2^{9}$	
	fx $\times 1/2^{10}$	
	fx $\times 1/2^{13}$	
	fx $\times 1/2^{15}$	

fpll-div: High-speed clock for peripheral function
fx: Low-speed clock for peripheral function
fs: System clock
*1 Can be used when fpll-div is selected as a clock source of time base timer
*2 Can be used when fx is selected as a clock source of time base timer

When changing the frequency of fpll-div by bp7 to bp4 of OSCCNT register, it should be performed after the time base timer / free-run timer function are stopped.

10.1.2 Block Diagram

■ Timer 6 and Time Base Timer Block Diagram

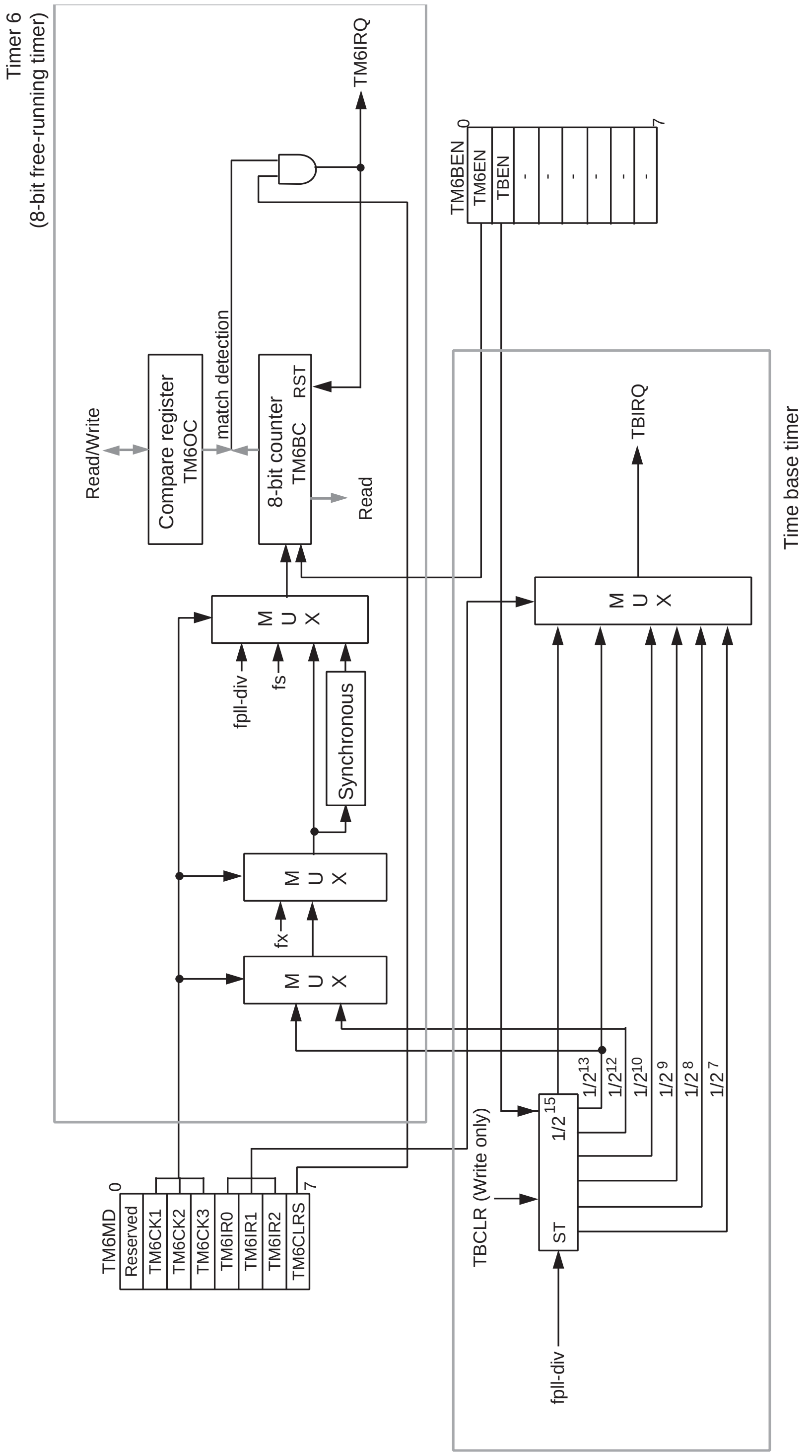


Figure:10.1.1 Block Diagram (Timer 6 and Time Base Timer)

10.2 Control Registers

Timer 6 consists of binary counter (TM6BC), compare register (TM6OC), and is controlled by mode register (TM6MD). Time base timer is controlled by mode register (TM6MD) and time base timer clear register (TBCLR). Both timers are operated by an enable signal of the timer 6 enable register (TM6BEN).

10.2.1 Control Registers

Table:10.2.1 shows the registers that control timer 6, time base timer.

Table:10.2.1 Control Registers

Table remarks √: With function -: Without function

	Register	Address	R/W	Function	Page	MN101EFA8	MN101EFA3	MN101EFA7	MN101EFA2
Timer 6	TM6BC	0x03F78	R	Timer 6 binary counter	X-7	√	√	√	√
	TM6OC	0x03F79	R/W	Timer 6 compare register	X-7	√	√	√	√
	TM6MD	0x03F7A	R/W	Timer 6 mode register	X-9	√	√	√	√
	TM6BEN	0x03F7C	R/W	Timer 6 enable register	X-8	√	√	√	√
	TM6ICR	0x03FF3	R/W	Timer 6 interrupt control register	IV-22	√	√	√	√
Time base timer	TM6MD	0x03F7A	R/W	Timer 6 mode register	X-9	√	√	√	√
	TBCLR	0x03F7B	W	Time base timer clear control register	X-7	√	√	√	√
	TBICR	0x03FF4	R/W	Time base interrupt control register	IV-23	√	√	√	√

R/W: Readable/Writable
R: Read only
W: Write only

10.2.2 Programmable Timer Registers

Timer 6 is a 8-bit programmable counter.

Programmable counter consists of compare register (TM6OC) and binary counter (TM6BC).

Binary counter is a 8-bit up-counter. When the TM6CLRS flag of the timer 6 mode register (TM6MD) is "0" and the interrupt cycle data is written to the compare register (TM6OC), the timer 6 binary counter (TM6BC) is cleared to 0x00.

- Timer 6 Binary Counter (TM6BC: 0x03F78)

bp	7	6	5	4	3	2	1	0
Flag	TM6BC7	TM6BC6	TM6BC5	TM6BC4	TM6BC3	TM6BC2	TM6BC1	TM6BC0
At reset	0	0	0	0	0	0	0	0
Access	R	R	R	R	R	R	R	R

- Timer 6 Compare Register (TM6OC: 0x03F79)

bp	7	6	5	4	3	2	1	0
Flag	TM6OC7	TM6OC6	TM6OC5	TM6OC4	TM6OC3	TM6OC2	TM6OC1	TM6OC0
At reset	X	X	X	X	X	X	X	X
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Time base timer can be reset its operation by the software. Time base timer can be cleared by writing an arbitrary value to the time base timer clear control register (TBCLR).

- Time Base Timer Clear Control Register (TBCLR: 0x03F7B)

bp	7	6	5	4	3	2	1	0
Flag	TBCLR7	TBCLR6	TBCLR5	TBCLR4	TBCLR3	TBCLR2	TBCLR1	TBCLR0
At reset	-	-	-	-	-	-	-	-
Access	W	W	W	W	W	W	W	W

10.2.3 Timer 6 Enable Register

This register controls the starting operation of the timer 6 and the time base timer.

■ Timer 6 Enable Register (TM6BEN: 0x03F7C)

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	-	Reserved	TBEN	TM6EN
At reset	-	-	-	-	-	0	0	0
Access	-	-	-	-	-	R/W	R/W	R/W

bp	Flag	Description
7-3	-	-
2	Reserved	Always set to “0”
1	TBEN	Time base timer operation control 0: Disabled 1: Enabled
0	TM6EN	Timer 6 operation control 0: Disabled 1: Enabled

The TM6EN flag of the TM6BEN register must be set to "1" to start the timer 6 operation.

The the TBEN flag of the TM6BEN register must be set to "1" to start the time base timer operation.

10.2.4 Timer Mode Register

This is readable/writable register that controls timer 6 and time base timer.

- Timer 6 Mode Register (TM6MD: 0x03F7A)

bp	7	6	5	4	3	2	1	0
Flag	TM6CLRS	TM6IR2	TM6IR1	TM6IR0	TM6CK3	TM6CK2	TM6CK1	TM6CK0
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7	TM6CLRS	Timer 6 binary counter clear selection flag 0: TM6BC initialization enabled when writing to TM6OC. 1: TM6BC initialization disabled when writing to TM6OC. * TM6IRQ is disabled when TM6CLRS = 0, TM6IRQ is enabled when TM6CLRS = 1.
6-4	TM6IR2 TM6IR1 TM6IR0	Time base timer interrupt cycle selection 000: Time base selection clock × $1/2^7$ 001: Time base selection clock × $1/2^8$ 010: Time base selection clock × $1/2^9$ 011: Time base selection clock × $1/2^{10}$ 10X: Time base selection clock × $1/2^{13}$ 11X: Time base selection clock × $1/2^{15}$
3-1	TM6CK3 TM6CK2 TM6CK1	Timer 6 clock source selection 000: fpll-div 001: fs 010: fx 011: Synchronous fx 100: Time base selection clock × $1/2^{13}$ 101: Synchronous time base selection clock × $1/2^{13}$ 110: Time base selection clock × $1/2^{12}$ 111: Synchronous time base selection clock × $1/2^{12}$
0	TM6CK0	Time base timer clock source selection 0: fpll-div 1: fx

10.3 8-bit Free-run Timer

10.3.1 Operation

■ 8-bit Free-run Timer (Timer 6)

The generation cycle of the timer interrupt should be set in advance, by the set value of the compare register (TM6OC) and the clock source selection. When the binary counter (TM6BC) reaches the set value of the compare register, an interrupt request is generated at the next count clock and the binary counter is cleared to restart count up from 0x00.

Table:10.3.1 shows selectable clock source.

Table:10.3.1 Clock Source at Timer Operation (Timer 6)

Clock source	One count time		
	At fpll-div=10 MHz	At fpll-div=8.39 MHz	At fpll-div=2 MHz
fpll-div	100 ns	119.1 ns	500 ns
fx	30.5 μs	30.5 μs	30.5 μs
fs	200 ns	238.3 ns	1000 ns
fpll-div × $1/2^{12}$	409.6 μs	487.4 μs	2048 μs
fpll-div × $1/2^{13}$	819.2 μs	976.4 μs	4096 μs
fx × $1/2^{12}$	125.0 ms	125.0 ms	125.0 ms
fx × $1/2^{13}$	250 ms	250 ms	250 ms

fpll-div = 10 MHz, 8.39 MHz, 2 MHz
fx = 32.768 kHz
fs = fpll-div/2

■ 8-bit Free-run Timer as 1 Minute-timer, 1 Second-timer

Table:10.3.2 indicates selection of clock source and setting value for TM6OC register are as indicated in the table below, when 8-bit free-run timer is used as 1-minute timer and 1-second timer.

Table:10.3.2 1 Minute-timer, 1 Second-timer (Timer 6) Setup

Interrupt Generation Cycle	Clock source	TM6OC register
1 min.	fx × $1/2^{13}$	0xEF
1 sec.	fx × $1/2^{13}$	0x03

fx = 32.768 kHz

When a 1-minute timer is set in accordance with the table above, it can be used to adjust seconds as bp2 waveform frequency (cycle) of TM6BC is 1Hz (1 sec.).

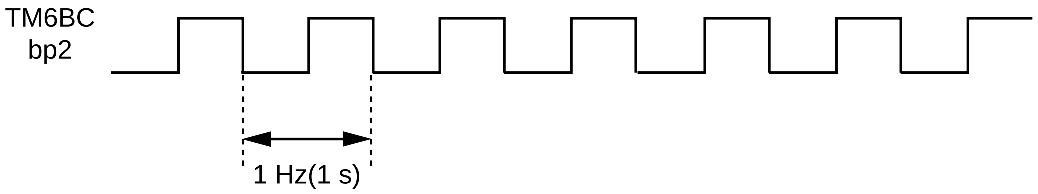


Figure:10.3.1 Waveform of TM6BC Register bp2 (Timer 6)

Switch the count clock after the timer operation is stopped, as the counting is not performed correctly during the timer operation.

■ Count Timing of Timer Operation (Timer 6)

Binary counter counts up with the selected clock source as a count clock.

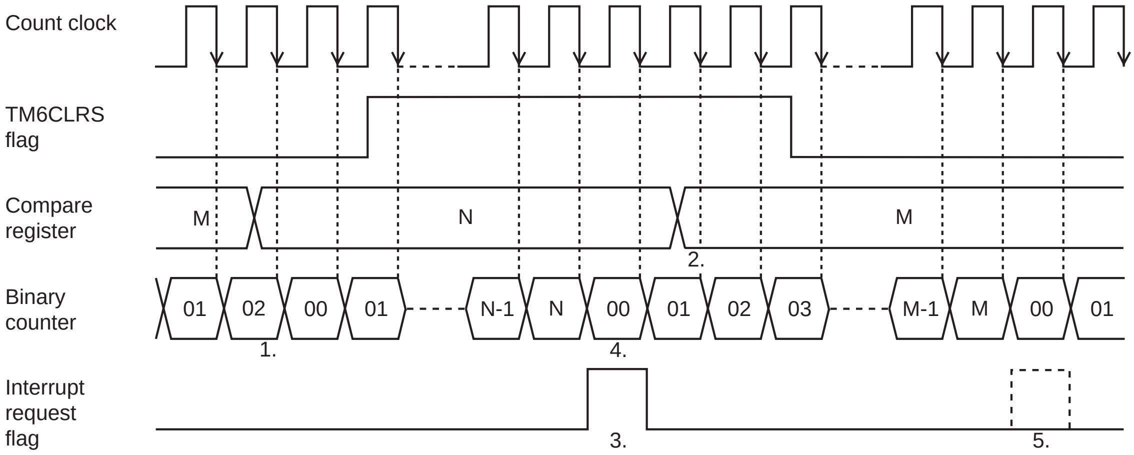


Figure:10.3.2 Count Timing of Timer Operation (Timer 6)

1. If any data is written to the compare register when the TM6CLRS flag is "0", the binary counter is cleared to 0x00.
2. Even if any data is written to the compare register when the TM6CLRS flag is "1", the binary counter is not cleared.
3. If the binary counter reaches the value of the compare register when the TM6CLRS flag is "1", an interrupt request flag is set at the next count clock.
4. When an interrupt request flag is set, the binary counter is cleared to 0x00 and restarts the counting.
5. Even if the binary counter reaches the value of the compare register when the TM6CLRS flag is "0", no interrupt request flag is set.

When the binary counter reaches the value in the compare register, the interrupt request flag is set and the binary counter is cleared at the next count clock.
So set the compare register as:
(Compare register setting) = (count till the interrupt request - 1)

If fx input is selected as a clock source in Timer 6, a uncertain value may be read when the binary counter is read during the operation.
To prevent this, select the synchronous fx input.

If fx is used as a clock source, the binary counter should be cleared before starting the timer operation. Also, to set 0x00 to the compare register, the synchronous fx should be used.

If the smaller value than the binary counter is set to the compare register during the counting operation, the binary counter continues counting till overflow. (When TM6CLRS flag is “1”.)

Up to 3 system clocks are needed from the timer n interrupt request flag till the next interrupt request flag. During the period, no interrupt request flag is generated even if a compare match occurs.

When the other is used, it is counted at "rising" of the count clock.

When fx is used as a clock source, it is counted at "falling" of the count clock. When the other is used, it is counted at "rising" of the count clock.

Count clock source should be changed when the timer interrupt is disabled.

If the TM6CLRS flag of the TM6MD register is set to "0", TM6BC can be initialized at every rewriting of TM6OC register, but in that state, the timer 6 interrupt is disabled. To use the timer 6 interrupt, set the TM6CLRS flag to "1" after rewriting the TM6OC register.

On the timer 6 clock source selection, the clock source setup of time base timer is necessary to select the time base selection clock or the synchronous time base selection clock.

For Timer 6 and Timer Base Timer on this LSI, operation or stop of the binary counter is controlled by the signal sampling the value of the TMnEN flag by the count clock. Therefore, if fx is selected as a count clock source, use the binary counter with attention to the following two points:

1. When reading the binary counter value after timer halts, put the TMnEN flag down, wait 1 count cycle, and read the value. To read the value, program to read the binary counter multiple times. However, if reading the value without waiting for 1 count cycle, the read value is [count value - 1].

2. When halting the timer and changing its setting (clock selection, function switching, etc.), wait 1 count clock after the timer is disabled before setting the timer. If the setting is switched during timer operation, proper operation is not guaranteed.

10.3.2 Setup Example

■ Timer Operation Setup (Timer 6)

Timer 6 generates interrupts constantly for timer function. Interrupts are generated in every 250 dividing (25 μs) by selecting fs (fpll-div = 10 MHz at operation) as a clock source.

An example setup procedure, with a description of each step is shown below.

Setup Procedure	Description
(1) Enable the binary counter initialization TM6MD(0x03F7A) bp7: TM6CLRS =0	(1) Set the TM6CLRS flag of the timer 6 mode register (TM6MD) to "0". At the time, the initialization of the timer 6 binary counter (TM6BC) is enabled.
(2) Disable the interrupt TM6ICR(0x03FF3) bp1: TM6IE =0	(2) Set the TM6IE flag of the TM6ICR register to "0" to disable the interrupt.
(3) Select the clock source TM6MD(0x03F7A) bp3 to 1: TM6CK3 to 1 =001	(3) Clock source can be selected by the TM6CK3 to 1 flag of the TM6MD register. Actually, fs is selected.
(4) Set the interrupt generation cycle TM6OC(0x03F79) =0xF9	(4) Set the interrupt generation cycle to the timer 6 compare register (TM6OC). At that time, TM6BC is initialized to 0x00.
(5) Enable the interrupt request TM6MD(0x03F7A) bp7: TM6CLRS =1	(5) Set the TM6CLRS flag of the TM6MD register to "1" to enable the interrupt request generation.
(6) Set the interrupt level TM6ICR(0x03FF3) bp7 to 6: TM6LV1 to 0 =01	(6) Set the interrupt level by the TM6LV1 to 0 flag of the timer 6 interrupt control register (TM6ICR). If the interrupt request flag may be already set, clear them. [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup]
(7) Enable the interrupt TM6ICR(0x03FF3) bp1: TM6IE =1	(7) Set the TM6IE flag of the TM6ICR register to "1" to enable the interrupt.
(8) Start the TM6 operation TM6BEN(0x03F7C) bp0: TM6EN =1	(8) Set the TM6EN flag of the TM6BEN register to "1" to start the timer 6.

As TM6OC is set, TM6BC is initialized to 0x00.

When TM6BC matches TM6OC, the timer 6 interrupt request flag is set at the next count clock and TM6BC is cleared to 0x00 to restart counting.

10.4 Time Base Timer

10.4.1 Operation

■ Time Base Timer (Time Base Timer)

Interrupt is constantly generated by a selected clock source and a interrupt generation cycle. Table:10.4.1 shows the interrupt cycle is combination with the clock source;

Table:10.4.1 Selection of Time Base Timer Interrupt Generation Cycle

Selected clock source	Interrupt generation cycle	
fpll-div	fpll-div × 1/2^{7}	12.8 µs
	fpll-div × 1/2^{8}	25.6 µs
	fpll-div × 1/2^{9}	51.2 µs
	fpll-div × 1/2^{10}	102.4 µs
	fpll-div × 1/2^{13}	819.2 µs
	fpll-div × 1/2^{15}	3.27 ms
fx	fx × 1/2^{7}	3.9 ms
	fx × 1/2^{8}	7.8 ms
	fx × 1/2^{9}	15.6 ms
	fx × 1/2^{10}	31.2 ms
	fx × 1/2^{13}	250 ms
	fx × 1/2^{15}	1 sec.

fpll-div =10 MHz fx =32.768 kHz

■ Count Timing Timer Operation (Time Base Timer)

The counter counts up with the selected clock source as a counter clock.

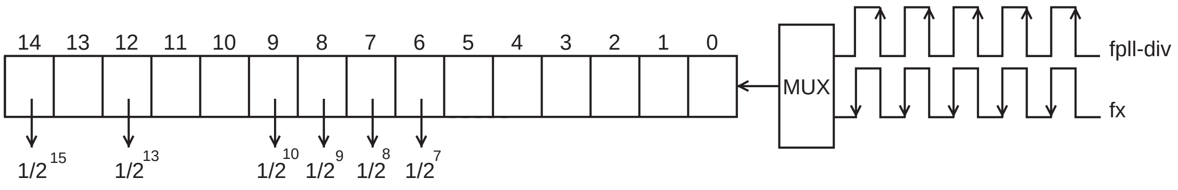


Figure:10.4.1 Count Timing of Timer Operation (Time Base Timer)

- When the selected interrupt cycle is passed, the interrupt request flag of the time base interrupt control register (TBICR) is set.

An interrupt may be generated at switching of the clock source. Enable the interrupt after switching the clock source.

The initialization can be done by writing an arbitrary value to the time base timer clear control register (TBCLR).

10.4.2 Setup Example

■ Timer Operation Setup (Time Base Timer)

An interrupt can be generated constantly with time base timer in the selected interrupt cycle. The interrupt generation cycle is fpll-div × $1/2^{13}$ (1 ms: fpll-div = 8.192 MHz) to generate interrupts.

An example setup procedure, with a description of each step is shown below

Setup Procedure	Description
(1) Select the clock source TM6MD(0x03F7A) bp0: TM6CK0 =0	(1) Select fpll-div as a clock source by the TM6CK0 flag of the timer 6 mode register (TM6MD).
(2) Disable the interrupt TBICR(0x03FF4) bp1: TBIE =0	(2) Set the TBIE flag of the TBICR register to "0" to disable the interrupt.
(3) Select the interrupt generation cycle TM6MD(0x03F7A) bp6 to 4: TM6IR2 to 0 =100	(3) Select the selected clock × $1/2^{13}$ as an interrupt generation cycle by the TM6IR2 to 0 flags of the TM6MD register.
(4) Initialize the time base timer TBCLR(0x03F7B) =0x00	(4) Write value to the time base timer clear control register (TBCLR) to initialize time base timer.
(5) Set the interrupt level TBICR(0x03FF4) bp7 to 6: TBLV1 to 0 =01	(5) Set the interrupt level by the TBLV1 to 0 flags of the time base interrupt control register (TBICR). If any interrupt request flag may be already set, clear them. [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup]
(6) Enable the interrupt TBICR(0x03FF4) bp1: TBIE =1	(6) Set the TBIE flag of the TBICR register to "1" to enable the interrupt.
(7) Start the time base timer operation TM6BEN(0x03F7C) bp1: TBEN =1	(7) Set the TBEN flag of the TM6BEN register to "1" to start the time base timer.

- When the selected interrupt generation cycle is passed, the interrupt request flag of the time base interrupt control register (TBICR) is set to "1".

Chapter 11 Watchdog Timer

11.1 Overview

This LSI has a watchdog timer. This watchdog timer is used to detect software processing errors. It is controlled by watchdog timer control register (WDCTR). Once the timer generates an overflow, watchdog timer interrupt (WDIRQ) is generated. After watchdog timer interrupts are generated two times in a row, software cannot execute in the intended sequence, thus forced reset is executed by the hardware.

11.1.1 Functions

Table:11.1.1 shows the functions of the watchdog timer.

Table:11.1.1 Watchdog Timer Functions

Watchdog time-out cycle setup selection	2^{16} × system clock cycle
	2^{18} × system clock cycle
	2^{20} × system clock cycle
Watchdog timer operation start timing selection	When the reset is released When “1” is written to WDEN flag

! To select the watchdog timer operation start timing, refer to [Chapter 2 2.2.7 Flash Option]. When the flash option is not set, “When the reset is released” is selected to start watchdog timer.

11.1.2 Block Diagram

■ Watchdog Timer Block Diagram

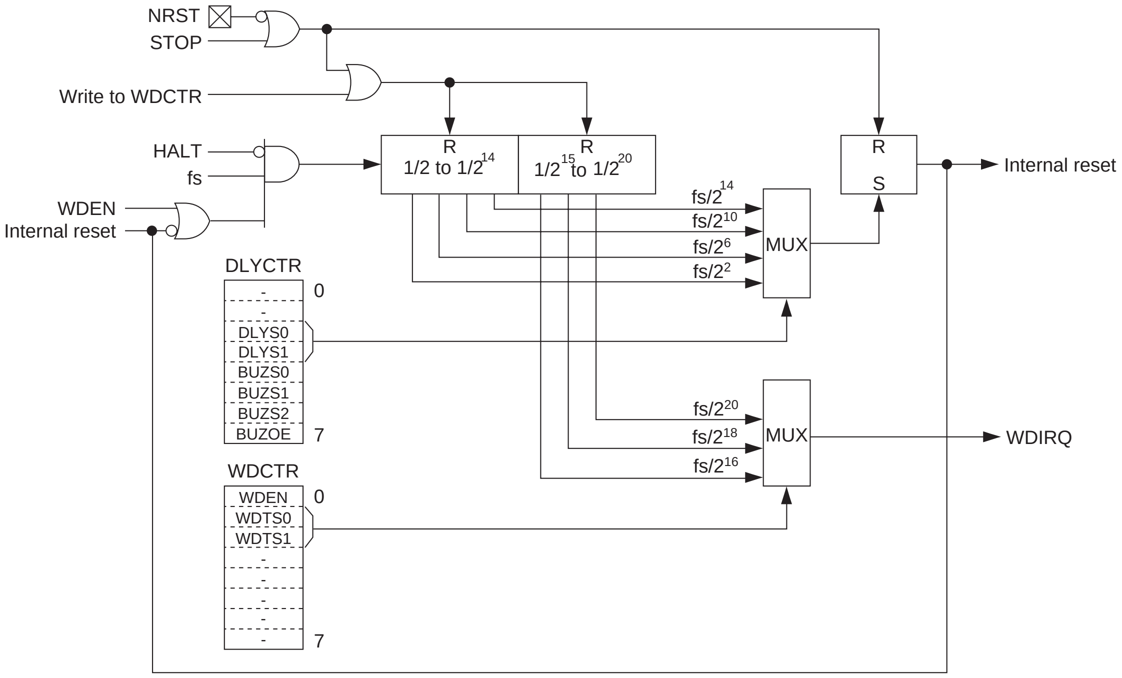


Figure:11.1.1 Watchdog Timer Block Diagram

Watchdog timer is also used as a timer to count oscillation stabilization wait time when recovering from STOP mode or releasing the reset. Otherwise, it is used as a timer to detect software processing errors.

Watchdog timer is initialized during reset or in STOP mode. It starts counting using system clock (fs) as a clock source from the initial value (0x0000). The oscillation stabilization wait time is set by oscillation stabilization wait time control register (DLYCTR).

11.2 Control Registers

The watchdog timer consists of control register (WDCTR).

11.2.1 Control Registers

Table:11.2.1 shows the registers that control watchdog timer.

Table:11.2.1 Watchdog Timer Control Register Functions

Table remarks √: With function -: Without function

Register	Address	R/W	Function	Page	MN101EF A8/A3	MN101EF A7/A2
WDCTR	0x03F02	R/W	Watchdog timer control register	XI-5	√	√
DLYCTR	0x03F03	R/W	Oscillation stabilization wait time control register	XI-6	√	√
PRTKEY	0x03E50	R/W	Register protect control register	XI-6	√	√

R/W: Readable/Writable

11.2.2 Watchdog Timer Control Register

Watchdog timer is controlled by watchdog timer control register (WDCTR).

■ Watchdog Timer Control Register (WDCTR: 0x03F02)

bp	7	6	5	4	3	2	1	0
Flag	-	-	Reserved	Reserved	Reserved	WDTS1	WDTS0	WDEN
At reset	-	-	0	0	0	1	1	*Note
Access	-	-	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7 to 6	-	-
5 to 3	Reserved	Always set to “0”.
2 to 1	WDTS1 WDTS0	Watchdog error detect cycle setup 00:2^{16} × system clock cycle 01: 2^{18} × system clock cycle 1X: 2^{20}× system clock cycle
0	WDEN	Watchdog timer enable (Refer to [Chapter 2 2.2.7 Flash Option]) 0: Disable watchdog timer. 1: Enable watchdog timer.

To change above registers, it is necessary to access the register protect control register (PRTKEY) to enable writing.

Set the watchdog timer error detect cycle to “2^{20} × system clock cycle”, when the flash memory is rewritten by the command library.

An initial value of WDEN flag of WDCTR register is determined based on a value of flash option. For the specific procedures, refer to [Chapter 11 11.2.3 Timing to Start Watchdog Timer].

■ Oscillation Stabilization Wait Time Control Register (DLYCTR: 0x03F03)

bp	7	6	5	4	3	2	1	0
Flag	BUZOE	BUZS2	BUZS1	BUZS0	DLYS1	DLYS0	-	-
At reset	0	0	0	0	0	1	-	-
Access	R/W	R/W	R/W	R/W	R/W	R/W	-	-

bp	Flag	Description
7	BUZOE	Buzzer output selection (Buzzer Functions) 0: Disable buzzer output 1: Enable buzzer output.
6 to 4	BUZS2 BUZS1 BUZS0	Buzzer output frequency (Buzzer Functions) 000: fpll-div / 2^{14} 001: fpll-div / 2^{13} 010: fpll-div / 2^{12} 011: fpll-div / 2^{11} 100: fpll-div / 2^{10} 101: fpll-div / 2^{9} 110: fx / 2^{4} 111: fx / 2^{3}
3 to 2	DLYS1 DLYS0	Oscillation stabilization wait period selection (Watchdog Timer Function) 00: 2^{14} × system clock cycle 01: 2^{10} × system clock cycle 10: 2^{6} × system clock cycle 11: 2^{2} × system clock cycle
1 to 0	-	-

For the oscillation stabilization wait cycle required for high-speed/low-speed oscillation by the settings of DLYS1 to DLYS0 flags, it is recommended to consult your oscillator manufacturer for determining the appropriate values.

When returning from STOP mode, 100 μs or longer of oscillation stabilization wait cycle must be set for the internal regulator to stabilize.

Refer to [Chapter 12 Buzzer] for buzzer function.

■ Register Protect Control Register (PRTKEY: 0x03E50)

The register protect control register enables the writing to the registers to be protected.
If the registers to be protected are accessed without writing a normal value to this register, it is judged an error writing and NMI interrupt is generated.

bp	7	6	5	4	3	2	1	0
Flag	PRTK7	PRTK6	PRTK5	PRTK4	PRTK3	PRTK2	PRTK1	PRTK0
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-0	PRTK7-0	Register protect control 0x44: Enable writing to WDCTR register Others: Disable writing to the registers to be protected

After the writing to the registers to be protected is enabled in PRTKEY register and that register is changed, it is recommended to write an arbitrary value to PRTKEY register to set the registers to be protected to “writing disable”.

We recommend that an interrupt is disabled between PRTKEY writing and appropriate register writing. If an interrupt is not be disable, save a value of PRTKEY at the front of the interrupt processing program, and recover it at the end of the program.

11.2.3 Timing to Start Watchdog Timer

This LSI allocates 0x040C1 of memory area as flash option areas.

When turning on the power or restarting from reset, refer to the flash option area to automatically read out the values set in hardware. Then set the timing to start watchdog timer. Refer to [Chapter 2 2.2.7 Flash Option].

The timing to start watchdog timer can be selected by setting the timing to start watchdog timer in the flash option area.

■ Flash Option 1 (FLOP1: 0x040C1)

bp	7	6	5	4	3	2	1	0
Flag	WDEN_INIT	-	-	-	-	-	-	-

bp	Flag	Description
7	WDEN_INIT	Select the timing to start watchdog timer. 1: When the reset is released 0: When "1" is written to WDEN flag
6 to 0	-	-

Values set to WDEN_INIT	Initial values of WDEN flag
0	1: Start watchdog time
1	0: Halt watchdog timer

Initial values of WDEN flag of watchdog timer control register (WDCTR) are determined by values which are set to WDEN_INIT flag of Flash Option 1. Initial values of WDEN flag are the same as the values set to the WDEN_INIT flag.

Flash option function is set when the power is turned on or restarted from reset. Therefore, rewriting is not reflected while microcomputer is working.

11.3 Operation

11.3.1 Operation

The watchdog timer counts using system clock (fs) as a clock source. When the counter of the watchdog timer overflows, the watchdog interrupt (WDIRQ) is generated as a non-maskable interrupt (NMI). When the LSI is reset, the watchdog timer is disabled. However, once the watchdog timer starts operating, it cannot be stopped except when the LSI is reset. Use watchdog timer control register (WDCTR) to disable the watchdog timer and to set the time-out period.

There are two options to set the timing to start watchdog timer using flash option: one is "When the reset is released" and another is "When "1" is written to WDEN flag". Once watchdog timer starts, it is impossible to halt the operation other than reset.

In the event that an watchdog interrupt (WDIRQ) is generated twice consecutively, the software cannot execute in the intended sequence, thus forced reset is executed by the hardware.

Once the watchdog timer starts operating, it cannot be stopped.
However, the timer stops when the CPU operates in HALT/STOP mode.

■ Usage of Watchdog Timer

When the watchdog timer function is used, it needs to be cleared in a certain period of time to prevent an overflow of the watchdog timer. In the event of software failure, the software cannot execute in the intended sequence, thus the watchdog timer will overflow due to the errors.

Programming of the watchdog is generally performed in the last step of its programming.

■ Detection of Incorrect Code Execution

Start watchdog timer so as to be cleared in the certain cycle while a program is properly executed.
In this LSI, the watchdog timer detects errors when the watchdog timer overflows.

When the watchdog timer detects any error, the watchdog interrupt (WDIRQ) is generated as a non-maskable interrupt (NMI).

■ How to Clear a Watchdog Timer

Watchdog timer can be cleared by writing any value to watchdog timer control register (WDCTR). It is recommended to use the bit set (BSET) instructions and etc., which do not change the value in WDCTR register.

■ Watchdog Time-out Period

The watchdog time-out period is decided by the WDTS1 and WDTS0 flags of WDCTR register and the system clock (fs). When the watchdog timer is not cleared before it reaches the set value, it is regarded as an error and the watchdog interrupt (WDIRQ) of the non-maskable interrupt (NMI) is generated.

The system clock is determined by CPU mode control register (CPUM).

Refer to [Chapter 3 3.2.2 Oscillation Control Registers].

The watchdog time-out period is generally decided based on an execution time of a main routine of a program. Be sure to set a time-out period whose value is longer than a value calculated by dividing an execution time of a main routine by a natural number (1, 2, 3,...).
And insert a command to clear the watchdog timer at regular intervals for the same times as the natural number.

■ Watchdog Timer and CPU OPERATION Mode

Table:11.3.1 shows the watchdog timer conditions based on each CPU mode.

Table:11.3.1 Watchdog Timer Condition in Each CPU OPERATION mode

CPU OPERATION mode	Watchdog timer condition
NORMAL	Counting up with the system clock. * The counting continues regardless of switching to NORMAL, SLOW or IDLE mode.
SLOW	
IDLE	
HALT	Counting stops. (the counting value is retained.)
STOP	Counting stops. (the counting value is cleared.) * Watchdog interrupts cannot be generated in STOP mode.
After returning from STOP mode	Counting continues after the oscillation stabilization wait time had passed and the watchdog timer is enabled. Counting stops after the oscillation stabilization wait time had passed and the watchdog timer is disabled.
After reset is released	Counting stops.

> ❗ When transit to STOP mode, the counter in the watchdog timer is cleared.

■ Protect Function of Watchdog Timer Control Register

When watchdog timer control registers are rewritten, it is necessary to access the register protect control register (PRTKEY) to enable rewriting to prevent the wrong writing by CPU processing errors.

■ Watchdog Timer Operation Start Timing Selection Function - WDEN_INIT: when the reset is released

Watchdog timer starts automatically after the reset is released. Watchdog timer is also used to count the oscillation stabilization wait time. For this, watchdog timer continues counting to detect any incorrect code execution without clearing binary counter after the oscillation stabilization wait time had passed.

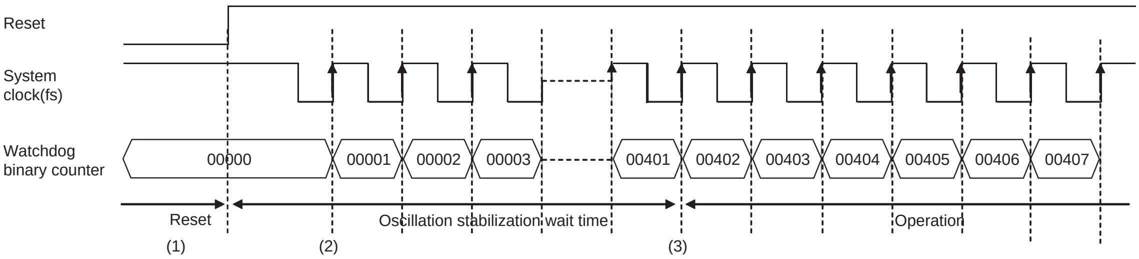


Figure:11.3.1 Operation (WDEN_INIT: when the reset is released)

(1) After the microcomputer is reset, binary counter of watchdog timer is cleared to 0x00000.

(2) After the reset is released, system clock starts its operation. Watchdog timer is also used as a timer to count the oscillation stabilization wait time. Therefore, the binary counter counts up at a rising edge of the system clock after the reset is released.

(3) Once the timer completes the counting of the oscillation stabilization wait time (system clock × 2^{10}) after the reset is released, the microcomputer starts its operation. Once the microcomputer starts, the watchdog counter continues counting to detect any incorrect code execution.

When WDEN_INIT is set to “1 (when the reset is released)”:
after the reset is released, watchdog timer needs to be cleared within a period of time, which is calculated by subtracting “the oscillation stabilization wait time” from “the watchdog time-out period”.

■ Watchdog Timer Operation Start Timing Selection Function - WDEN_INIT: when "1" is written to WDEN flag

After the reset is released, writing "1" to WDEN flag of watchdog timer control register (WDCTR) enables the watchdog timer. Be sure to set operation with software.

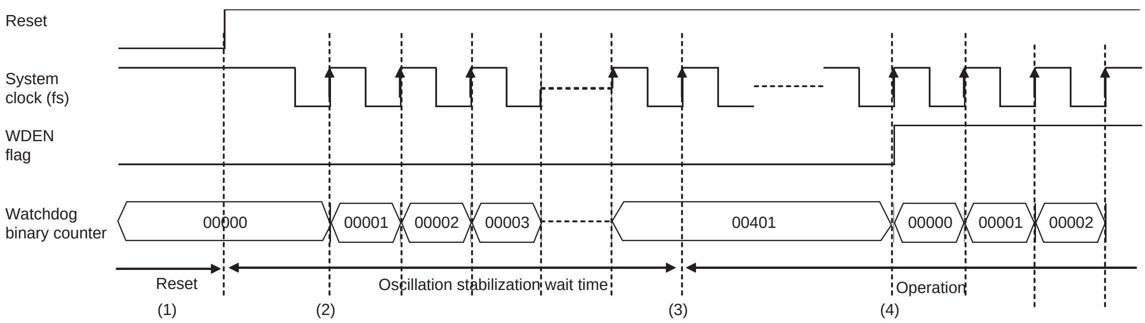


Figure:11.3.2 Operation (WDEN_INIT: when "1" is written to WDEN flag)

(1) After the microcomputer is reset, binary counter of the watchdog timer is cleared to 0x00000.

(2) After the reset is released, system clock starts its operation. Watchdog timer is also used as a timer to count the oscillation stabilization wait time. Therefore, the binary counter counts up at the rising edge of system clock after the reset is released.

(3) When the oscillation stabilization wait time (system clock × 2^{10}) had passed after the reset is released, microcomputer starts to operate. Simultaneously, the watchdog timer stops counting.

(4) Writing "1" to WDEN flag of watchdog timer control register (WDCTR), the binary counter is cleared to 0x00000, and the watchdog timer starts to operate. The binary counter counts up at the rising edge of the system clock.

■ Register Protect Control

The register protect function enables the writing by writing a normal value, allocated to the registers to be protected, to the register protect control register.
If the registers to be protected are written without writing a normal value to the register protect control register, NMI interrupt is generated.

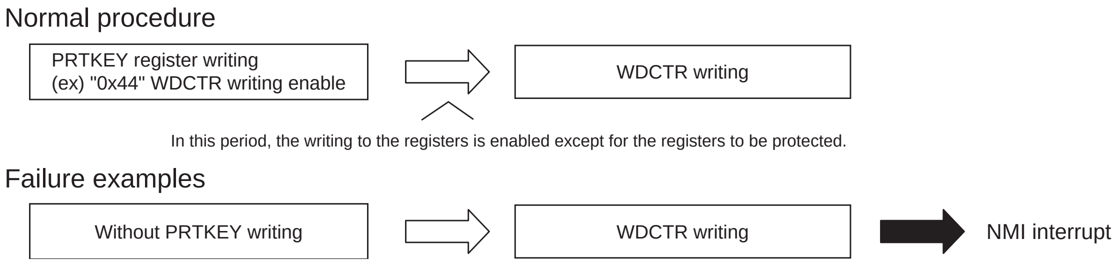

Normal procedure
PRTKEY register writing
(ex) "0x44" WDCTR writing enable
WDCTR writing
In this period, the writing to the registers is enabled except for the registers to be protected.
Failure examples
Without PRTKEY writing
WDCTR writing
NMI interrupt

Figure:11.3.3 Register Protect Operation

11.3.2 Setup Example

Watchdog timer has a function to detect errors. On the following example, the time-out period is set to 2^{18} × system clock. The following is an example of setup procedures.

■ Initial Setup Program (Example to Initialize Watchdog Timer)

Setup Procedure	Description
(1) Set the time to out period PRTKEY(0x03E50) bp7 to 0: PRTK7 to 0 =0x44 WDCTR(0x03F02) bp2 to 1: WDTS1 to 0 =01	(1) “0x44” must be written to PRTKEY register to enable writing. Set the WDTS1 to 0 flags of WDCTR register to "01" to select the time-out period to 2^{18} × system clock.
(2) Start the watchdog timer operation PRTKEY(0x03E50) bp7 to 0: PRTK7 to 0 =0x44 WDCTR(0x03F02) bp0: WDEN =1	(2) “0x44” must be written to PRTKEY register to enable writing. Set the WDEN flag of WDCTR register to “1” to start the watchdog timer operation.
(3) Set the register protect PRTKEY(0x03E50) bp7 to 0: PRTK7 to 0 =0xFF	(3) Set any value (0xFF) to PRTKEY register to set the protect again.

■ Main Routine Program (Example to Clear Watchdog Timer Periodically)

Setup Procedure	Description
(1) Set the watchdog timer for the constant clear Writing to WDCTR(0x03F02) PRTKEY(0x03E50) bp7 to 0: PRTK7 to 0 =0x44 (c.f.) BSET (WDCTR) WDEN (bp0: WDEN=1) PRTKEY(0x03E50) bp7 to 0: PRTK7 to 0 =0xFF	(1) “0x44” must be written to PRTKEY register to enable writing. Clear the watchdog timer by the cycle from 2^{18} × system clock. The watchdog timer clear should be inserted in the main routine, with the same cycle, and to be the set cycle. It is recommended to use the bit set (BSET) instructions and etc., which do not change the value in WDCTR register. Set any value (0xFF) to PRTKEY register to set the protect again.

■ Interrupt Service Routine Setup

Setup Procedure	Description
(1) Set the watchdog interrupt service routine NMICR(0x03FE1) TBNZ (NMICR) WDIR, WDPR0	(1) If the watchdog timer overflows, the non-maskable interrupt is generated. Confirm that the WDIR flag of the non-maskable interrupt control register (NMICR) is "1" in the interrupt processing routine and execute the appropriate processing for the system.

The operation just before the watchdog interrupt may have been executed wrongly. In that case, proper operation is not guaranteed.

Chapter 12 Buzzer

12

12.1 Overview

This LSI has a buzzer. It can output the square wave that multiply by $1/2^9$ to $1/2^{14}$ of the high frequency oscillation clock.
Pins can be switched to BUZZERA/NBUZZERA, BUZZERB/NBUZZERB.

Table:12.1.1 Buzzer Pin Functions

Table remarks √: With function -: Without function

Functions	Pin Name	MN101EFA8	MN101EFA3	MN101EFA7	MN101EFA2
BUZZERA	P57	√	√	√	√
BUZZERB	P87	√	√	√	√
NBUZZERA	P56	√	√	√	√
NBUZZERB	P86	√	√	√	√

In this manual, if there is not much functional difference in pins A and B, “A” and “B” of the pin names are omitted.

12.1.1 Functions

Table:12.1.2 shows the buzzer functions.

Table:12.1.2 Buzzer Functions

Output selection	Disable buzzer output
	Enable buzzer output
Buzzer reverse output	Polarity reverse output
Output pin selection	BUZZERA, NBUZZERA output
	BUZZERB, NBUZZERB output
Buzzer output frequency selection	fpll-div/2^{14}
	fpll-div/2^{13}
	fpll-div/2^{12}
	fpll-div/2^{11}
	fpll-div/2^{10}
	fpll-div/2^{9}
	fx /2^{4}
	fx /2^{3}
Oscillation stabilization wait cycle selection	fs/2^{14}
	fs/2^{10}
	fs/2^{6} *1
	fs/2^{2} *1

*1: Use this function at low-speed operation (SLOW mode), not at high-speed operation (NORMAL mode).

At the instant that the BUZOE flag is set to "0", the output of the buzzer becomes "Low".

When changing the frequency of fpll-div by bp7 to bp4 of OSCCNT register, it should be executed after the buzzer function is stopped.

12.1.2 Block Diagram

■ Buzzer Block Diagram

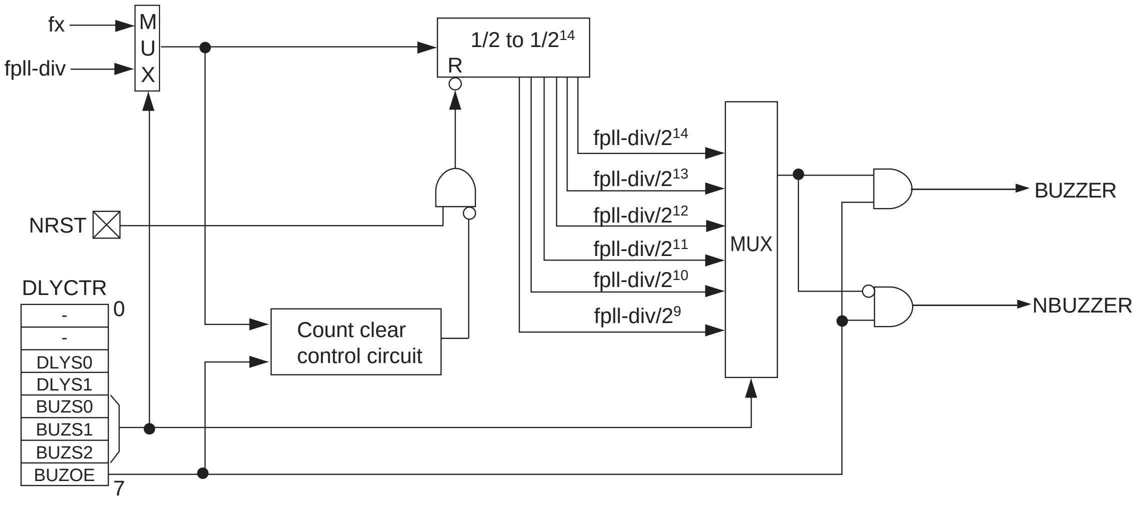


Figure:12.1.1 Buzzer Block Diagram

12.2 Control Register

Buzzer function is controlled by port control registers and upper 4 bits of oscillation stabilization wait time control register (DLYCTR).

12.2.1 Registers

Table:12.2.1 shows the registers that control buzzer function.

Table:12.2.1 Buzzer Control Register

Table remarks √: With function -: Without function

Register	Address	R/W	Function	Page	MN101 EFA8	MN101 EFA3	MN101 EFA7	MN101 EFA2
DLYCTR	0x03F03	R/W	Oscillation stabilization wait time control register	XII-6	√	√	√	√
P5DIR	0x03F95	R/W	Port 5 direction control register	V-46	√	√	√	√
P5OMD	0x03EB5	R/W	Port 5 output mode register	V-47	√	√	√	√
P8DIR	0x03F98	R/W	Port 8 direction control register	V-86	√	√	√	√
P8OMD	0x03EB8	R/W	Port 8 output mode register	V-87	√	√	√	√

R/W: Readable/Writable

In this manual, if there is not much functional difference in pins A and B, “A” and “B” of the pin names are omitted.

12.2.2 Oscillation Stabilization Wait Time Control Register

- Oscillation Stabilization Wait Time Control Register (DLYCTR: 0x03F03)

bp	7	6	5	4	3	2	1	0
Flag	BUZOE	BUZS2	BUZS1	BUZS0	DLYS1	DLYS0	-	-
At reset	0	0	0	0	0	1	-	-
Access	R/W	R/W	R/W	R/W	R/W	R/W	-	-

bp	Flag	Description
7	BUZOE	Buzzer output selection 0: Buzzer output disable 1: Buzzer output enable
6-4	BUZS2-0	Buzzer output frequency selection 000: fpll-div/2^{14} 001: fpll-div/2^{13} 010: fpll-div/2^{12} 011: fpll-div/2^{11} 100: fpll-div/2^{10} 101: fpll-div/2^{9} 110: fx /2^{4} 111: fx /2^{3}
3-2	DLYS1-0	Oscillation stabilization wait period selection (Watchdog Timer Functions) 00: 2^{14} × system clock cycle 01: 2^{10} × system clock cycle 10: 2^{6} × system clock cycle 11: 2^{2} × system clock cycle
1-0	-	-

Do not set BUZOE flag and BUZS2 to BUZS0 flags simultaneously.

The DLYS1 to DLYS0 flags are the setting flags for Watchdog timer.
Refer to [Chapter 11 Watchdog Timer] for details.

12.3 Operation

12.3.1 Operation

■ Buzzer

Buzzer outputs square wave with frequency $1/2^9$ to $1/2^{14}$ of the high oscillation clock (fpll-div). The BUZS2 to BUZS0 flags of DLYCTR register set the frequency of the buzzer output. The BUZOE flag of DLYCTR register enables/disables the buzzer output.

■ Buzzer Output Frequency

The frequency of buzzer output is determined by the frequency of oscillation clock and the BUZS2 to BUZS0 flags of DLYCTR register.

Table:12.3.1 Buzzer Output Frequency

fpll-div	fx	BUZS2	BUZS1	BUZS0	Buzzer output frequency
10 MHz	-	0	1	0	2.44 kHz
10 MHz	-	0	1	1	4.88 kHz
4 MHz	-	0	1	1	1.95 kHz
4 MHz	-	1	0	0	3.91 kHz
-	32 kHz	1	1	0	2 kHz
-	32 kHz	1	1	1	4 kHz

Immediately after the BUZOE flag is set to “0”, the buzzer output of both BUZZER and NBUZZER is turned “Low”.

12.3.2 Setup Example

■ Setup Example

With buzzer function, the square wave of 2.44 kHz is output from pin P57. This function uses 10 MHz as the high speed oscillation clock (fpll-div).

The following is an example of setup procedures.

Setup Procedure	Description
(1) Set the buzzer frequency DLYCTR (0x03F03) bp6 to 4: BUZS2 to 0 =010	(1) Set the BUZS2 to BUZS0 flag of DLYCTR register to "010" to select fpll-div/2^{12} to the buzzer frequency. When the high speed oscillation clock fpll-div is 10 MHz, the buzzer output frequency is 2.44 kHz.
(2) Set pin P57 P5OMD (0x03EB5) bp7: P5OMD7 =1 P5DIR (0x03F95) bp7: P5DIR7 =1	(2) Set the P5OMD7 flag of P5OMD register to "1" to set pin P57 to the special function. Set the P5DIR7 flag of P5DIR register to "1" to set output mode; then, “Low-level” is output from pin P57.
(3) Set Buzzer output ON DLYCTR (0x03F03) bp7: BUZOE =1	(3) Set the BUZSE flag of DLYCTR register to "1" to output the square wave of the setting frequency from pin P57.
(4) Set Buzzer output OFF DLYCTR (0x03F03) bp7: BUZOE =0	(4) Set the BUZOE flag of DLYCTR register to "0" to disable the buzzer, and P57 outputs “Low- level”.

Be sure to set the buzzer output to “enable” after setting the buzzer frequency.

Chapter 13 Serial Interface

13.1 Overview

This LSI has 4 serial interfaces (serial 0, 1, 2 and 4).

Serial interfaces 0, 1 and 2 can be used for clock synchronous and UART (duplex) communication.
Serial interface 4 can be used for clock synchronous and IIC (multi-master) communication.

Table:13.1.1 Serial Interface Communication Types

	Serial Interface 0	Serial Interface 1	Serial Interface 2	Serial Interface 4
Clock synchronous	√	√	√	-
UART(duplex)	√	√	√	√
Multi master IIC	-	-	-	√

Table:13.1.2 shows the pins used for serial interface. Serial interfaces can switch pins to A-type or B-type. For A-type pin, "A" is added to the end of its name. For B-type pin, "B" is added to the end of its name.

Table:13.1.2 MN101EFA8/A3 Serial Interface Pins

		Serial Interface 0		Serial Interface 1		Serial Interface 2	Serial Interface 4	
Pin switching function		√		√		-	√	
Clock synchro-nous	Data I/O pin	SBO0A (P03)	SBO0B (P43)	SBO1A (P50)	SBO1B (P75)	SBO2 (P65)	SBO4A (P71)	SBO4B (P33)
	Data input pin	SBI0A (P02)	SBI0B (P44)	SBI1A (P51)	SBI1B (P76)	SBI2 (P66)	SBI4A (P70)	SBI4B (P35)
	Clock I/O pin	SBT0A (P04)	SBT0B (P45)	SBT1A (P52)	SBT1B (P77)	SBT2 (P67)	SBT4A (P72)	SBT4B (P34)
UART(duplex)	Data I/O pin	TXD0A (P03)	TXD0B (P43)	TXD1A (P50)	TXD1B (P75)	TXD2 (P65)	-	
	Data input pin	RXD0A (P02)	RXD0B (P44)	RXD1A (P51)	RXD1B (P76)	RXD2 (P66)	-	
Multi master IIC	Data I/O pin	-		-		-	SDA4A (P71)	SDA4B (P33)
	Clock I/O pin	-		-		-	SCL4A (P72)	SCL4B (P34)

Table:13.1.3 MN101EFA7/A2 Serial Interface Pins

		Serial Interface 0	Serial Interface 1		Serial Interface 2	Serial Interface 4
Pin switching function		-	√		-	-
Clock synchronous	Data I/O pin	SBO0A (P03)	SBO1A (P50)	SBO1B (P75)	SBO2 (P65)	SBO4A (P71)
	Data input pin	SBI0A (P02)	SBI1A (P51)	SBI1B (P76)	SBI2 (P66)	SBI4A (P70)
	Clock I/O pin	SBT0A (P04)	SBT1A (P52)	SBT1B (P77)	SBT2 (P67)	SBT4A (P72)
UART(duplex)	Data I/O pin	TXD0A (P03)	TXD1A (P50)	TXD1B (P75)	TXD2 (P65)	-
	Data input pin	RXD0A (P02)	RXD1A (P51)	RXD1B (P76)	RXD2 (P66)	-
Multi master IIC	Data I/O pin	-	-	-	-	SDA4A (P71)
	Clock I/O pin	-	-	-	-	SCL4A (P72)

In the text, if there is not much functional difference in pins A or B, “A” and “B” of the pin names are omitted.

Be sure to change the frequency of fpll-div by bp7 to bp4 of OSCCNT register after stopping the serial interface functions.

13.1.1 Functions

Table:13.1.4 shows clock synchronous serial interface functions.
Table:13.1.5 shows UART (duplex) serial interface functions.
Table:13.1.6 shows multi master IIC interface functions.

Set the transfer rate which is slower than the system clock (fs).

Table:13.1.4 Clock Synchronous Serial Interface Functions

	Serial Interface 0	Serial Interface 1	Serial Interface 2	Serial Interface 4
Interrupt	SC0TIRQ	SC1TIRQ	SC2TIRQ	SC4IRQ
	SC0RIRQ	SC1RIRQ	SC2RIRQ	
PIns to be used	SBO0	SBO1	SBO2	SBO4
	SBI0	SBI1	SBI2	SBI4
	SBT0	SBT1	SBT2	SBT4
3-wire system	√	√	√	√
2-wire system	√ (SBO0,SBT0)	√ (SBO1,SBT1)	√ (SBO2,SBT2)	√ (SBO4,SBT4)
Specification of transfer bit count (2 to 8 bits)	√	√	√	√
Selection of start condition	√	√	√	√
Specification of the first transfer bit	√	√	√	√
Input edge/output edge	√	√	√	√
SBO output control after transmission (High/Low/Last data hold)	√	√	√	√
Communication function in STANDBY mode (only slave reception is available)	√	√	√	√
Continuous operation	√	√	√	√
Selection of transfer clock dividing	√	√	√	-
Selection of transfer clock division ratio	No dividing	No dividing	No dividing	-
	Divide by 8	Divide by 8	Divide by 8	
	Divide by16	Divide by16	Divide by16	
Clock source	fpll-div/2	fpll-div/2	fpll-div/2	fpll-div/2
	fpll-div/4	fpll-div/4	fpll-div/4	fpll-div/4
	fpll-div/16	fpll-div/16	fpll-div/16	fpll-div/16
	fpll-div/64	fpll-div/64	fpll-div/64	fpll-div/64
	fs/2	fs/2	fs/2	fs/2
	fs/4	fs/4	fs/4	fs/4
	External clock	External clock	External clock	External clock
	Timer 0 to 3 or A	Timer 0 to 3 or A	Timer 0 to 3 or A	Timer 0 to 3 or A
	output divided by 1, 2, 4, 8 or 16	output divided by 1, 2, 4, 8 or 16	output divided by 1, 2, 4, 8 or 16	output divided by 1, 2, 4, 8 or 16
Maximum transfer rate	5.0 MHz	5.0 MHz	5.0 MHz	5.0 MHz

fpll-div: high-speed clock for peripheral functions
fs: system clock

Table:13.1.5 UART (duplex) Serial Interface Functions

		Serial Interface0	Serial Interface1	Serial Interface2
Interrupt	(at transmission completion)	SC0TIRQ	SC1TIRQ	SC2TIRQ
	(at reception completion)	SC0RIRQ	SC1RIRQ	SC2RIRQ
Pins to be used		TXD0	TXD1	TXD2
		RXD0	RXD1	RXD2
2-wire communication		√		
1-wire communication		√(TXD0)	√(TXD1)	√(TXD2)
Specification of transfer bit count/frame selection		7 bits + 1STOP		
		7 bits + 2STOP		
		8 bits + 1STOP		
		8 bits + 2STOP		
Selection of parity bit		√		
Parity bit control		0 parity		
		1 parity		
		Odd parity		
		Even parity		
Specification of the first transfer bit		√		
Selection of transfer clock dividing ratio		Divide by 8 / divide by16		
Clock source		fpll-div/2		
		fpll-div/4		
		fpll-div/16		
		fpll-div/64		
		fs/2		
		fs/4		
		Timer 0 to 3 or A		
		output divided by 1, 2, 4, 8 or 16		
Maximum transfer rate		300 kbps		

fpll-div: high-speed clock for peripheral functions
fs: system clock
In UART communication, a transfer clock is obtained by dividing a clock source internally.

Table:13.1.6 Multi Master IIC Serial Interface Functions

		Serial Interface 4
Interrupt		SC4IRQ (communication completion interrupt)
		SC4STPCIRQ (stop condition detection interrupt)
Pins to be used		SDA4
		SCL4
Communication mode	Master transmission	√
	Master reception	√
	Slave transmission	√
	Slave reception	√
Transfer format	Addressing format	√
	Free data format	Available only in master communication
Address format	7-bit address	√
	10-bit address	Available only in master transmission/reception or slave reception
	General call	√
Communication format	Standard mode (100 bps)	√
	High-speed mode (400 bps)	√
Specification of first transfer bit		2 to 8 bits (in master communication)
		8 bits (in slave communication)
Specification of transfer first bit		√
ACK bit selection		Available only in master communication
ACK bit level selection		√
Maximum transfer rate		Normal mode: 100 kHz
		High-speed mode: 400 kHz
Clock sources		fpll-div/2
		fpll-div/4
		fpll-div/16
		fpll-div/32
		fs/2
		fs/4
		Timer 0 to 3 or A
		output divided by 1, 2, 4, 8 or 16

fpll-div: high-speed clock for peripheral functions
fs: system clock
In IIC communication, a transfer clock is obtained by dividing a clock by 8.

13.1.2 Block Diagram

■ Serial Interface 0 Block Diagram

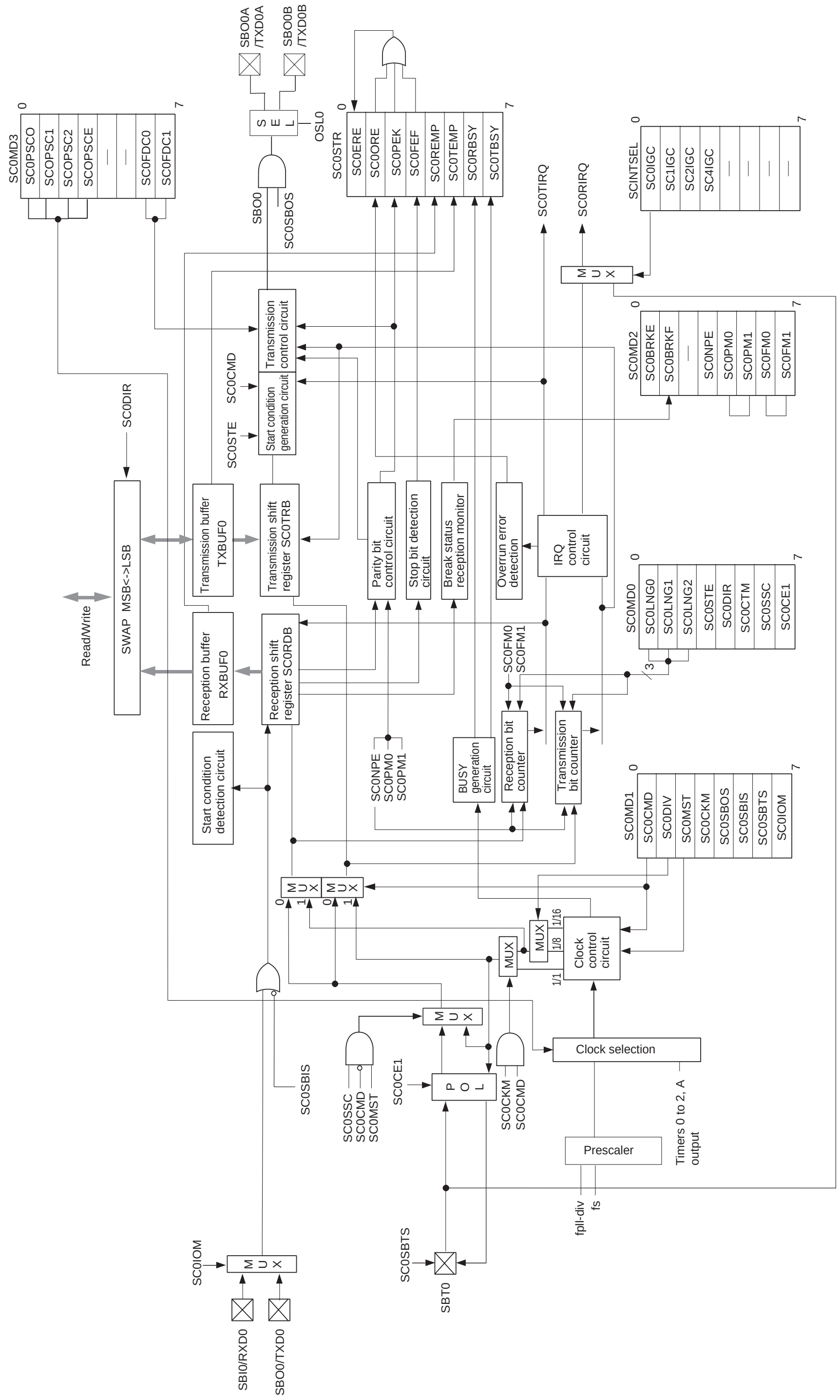


Figure:13.1.1 Serial Interface 0 Block Diagram

■ Serial Interface 1 Block Diagram

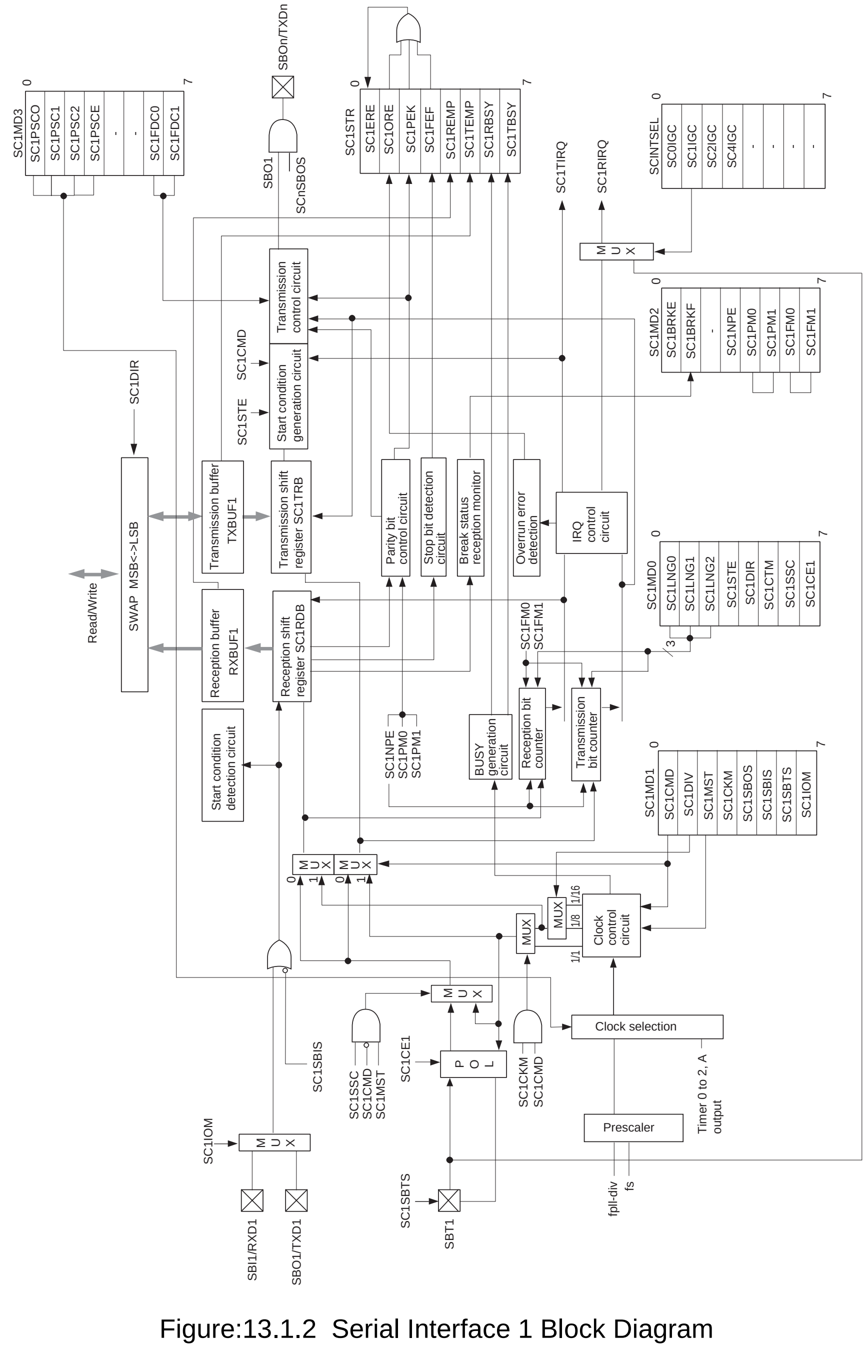


Figure:13.1.2 Serial Interface 1 Block Diagram

■ Serial Interface 2 Block Diagram

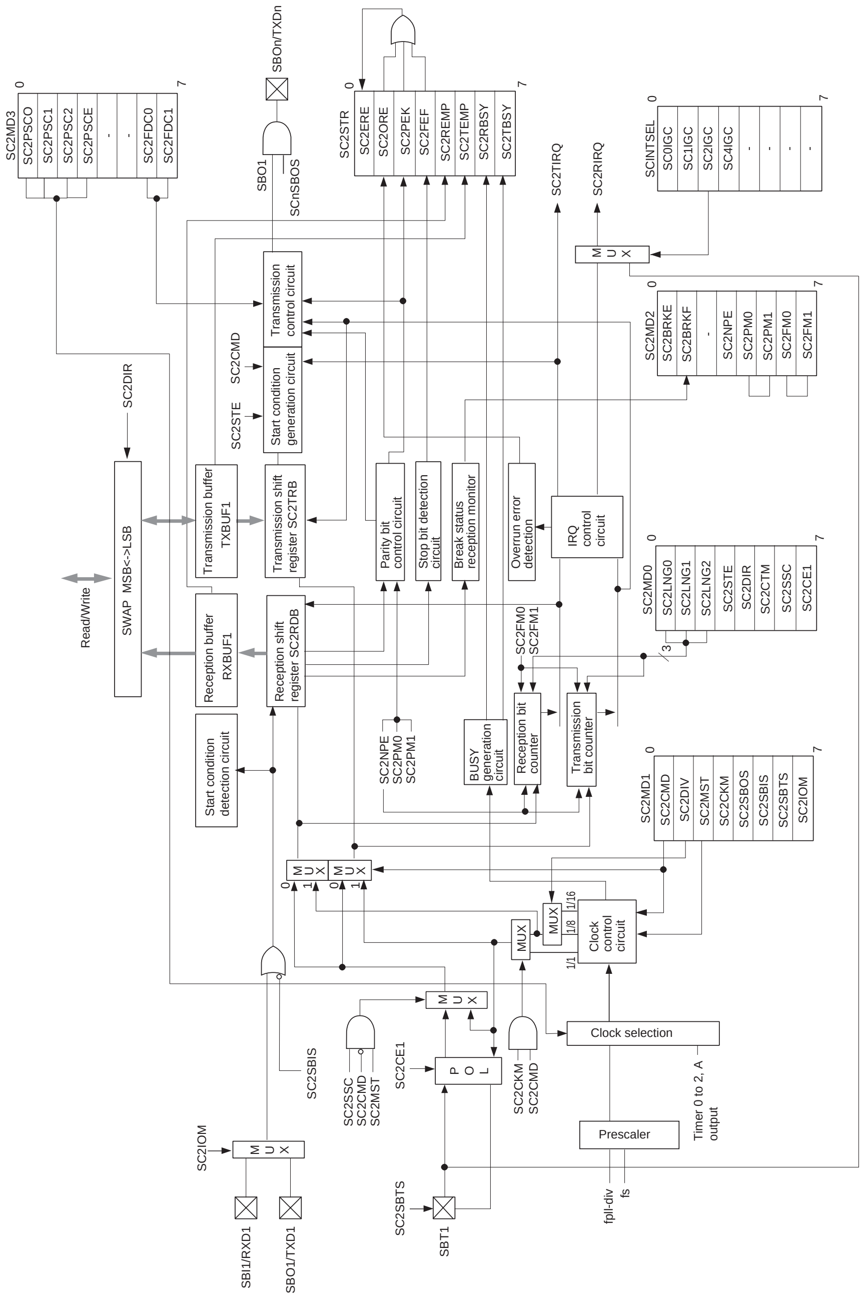


Figure:13.1.3 Serial Interface 2 Block Diagram

■ Serial Interface 4 Block Diagram

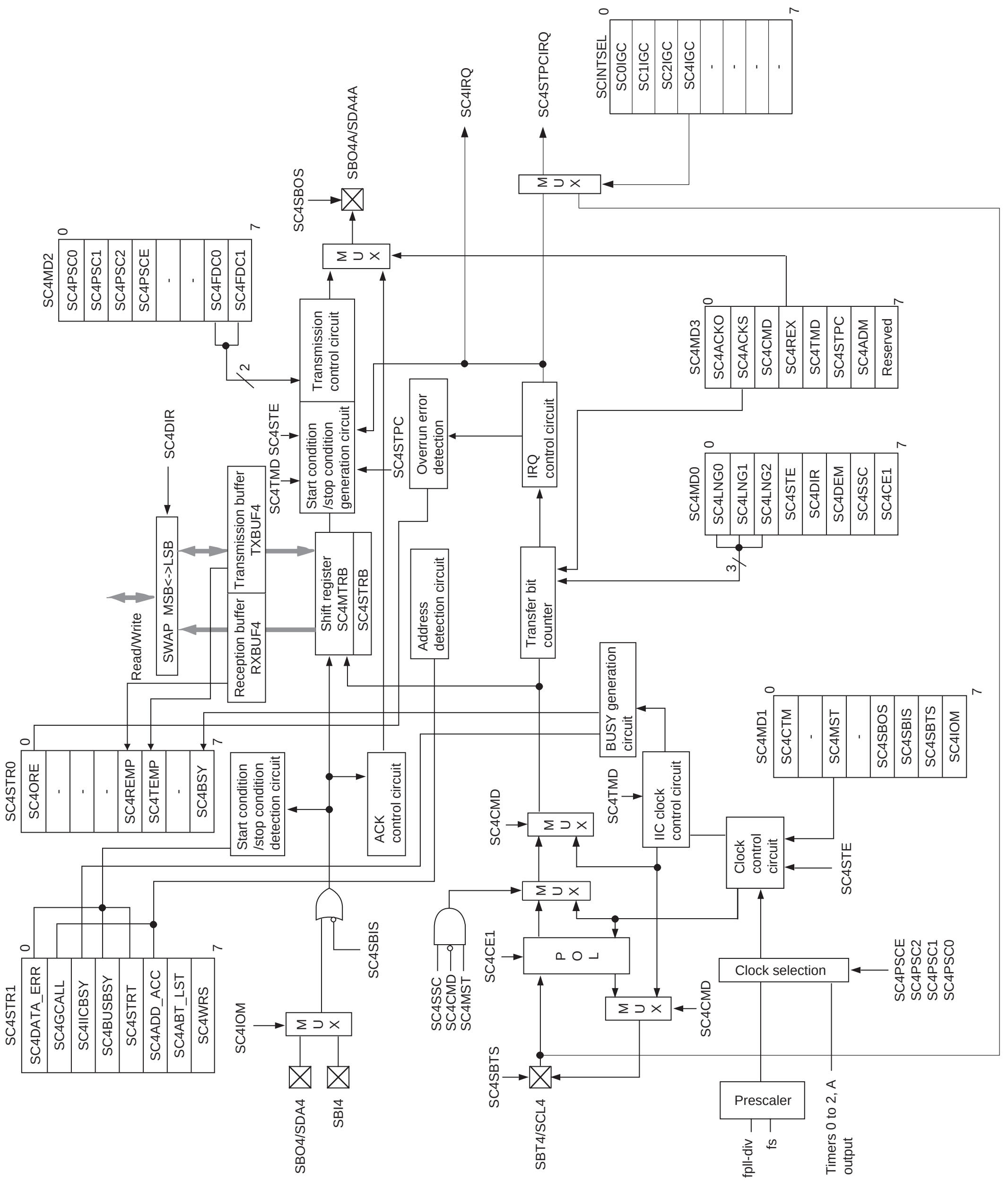


Figure:13.1.4 Serial Interface 4 Block Diagram

13.2 Control Registers

13.2.1 Registers

Table:13.2.1 shows registers to control serial interface.

To change setting values of mode registers, set forced reset first for serial interface by setting both SCnSBIS and SCnSBOS flags of SCnMD1 to “0”. (n = 0 to 2)

Table:13.2.1 Serial Interface Control Registers

Table remarks √: With function -: Without function

	Register	Address	R/W	Function	Page	MN101 EFA8	MN101 EFA3	MN101 EFA7	MN101 EFA2
Serial 0	SC0SEL	0x03F10	R/W	Serial 0 I/O pin switching control register	XIII-15	√	√	√	√
	SC0MD0	0x03F11	R/W	Serial interface 0 mode register 0	XIII-22	√	√	√	√
	SC0MD1	0x03F12	R/W	Serial interface 0 mode register 1	XIII-23	√	√	√	√
	SC0MD2	0x03F13	R/W	Serial interface 0 mode register 2	XIII-24	√	√	√	√
	SC0MD3	0x03F14	R/W	Serial interface 0 mode register 3	XIII-25	√	√	√	√
	SC0STR	0x03F15	R	Serial interface 0 status register	XIII-26	√	√	√	√
	RXBUF0	0x03F16	R	Serial interface 0 reception data buffer	XIII-21	√	√	√	√
	TXBUF0	0x03F17	R/W	Serial interface 0 transmission data buffer	XIII-21	√	√	√	√
	P0ODC	0x03EF0	R/W	Port 0 Nch open-drain control register	V-13	√	√	√	√
	P0DIR	0x03E90	R/W	Port 0 direction control register	V-10	√	√	√	√
	P0PLU	0x03EA0	R/W	Port 0 pull-up resistor control register	V-11	√	√	√	√
	P4ODC	0x03EF4	R/W	Port 4 Nch open-drain control register	V-40	√	√	-	-
	P4DIR	0x03E94	R/W	Port 4 direction control register	V-39	√	√	-	-
	P4PLUD	0x03EA4	R/W	Port 4 pull-up/pull-down resistor control register	V-39	√	√	-	-
	SC0TICR	0x03FFB	R/W	Serial 0 transmission interrupt control register	IV-28	√	√	√	√
	SC0RICR	0x03FFC	R/W	Serial 0 reception interrupt control register	IV-29	√	√	√	√
	SCINTSEL	0x03F28	R/W	Clock synchronous serial reception interrupt control register	XIII-27	√	√	√	√

R/W: Readable/Writable
R: Read only

Table remarks √: With function -: Without function

	Register	Address	R/W	Function	Page	MN101 EFA8	MN101 EFA3	MN101 EFA7	MN101 EFA2
Serial 1	SC1SEL	0x03F20	R/W	Serial 1 I/O pin switching control register	XIII-17	√	√	√	√
	SC1MD0	0x03F21	R/W	Serial interface 1 mode register 0	XIII-22	√	√	√	√
	SC1MD1	0x03F22	R/W	Serial interface 1 mode register 1	XIII-23	√	√	√	√
	SC1MD2	0x03F23	R/W	Serial interface 1 mode register 2	XIII-24	√	√	√	√
	SC1MD3	0x03F24	R/W	Serial interface 1 mode register 3	XIII-25	√	√	√	√
	SC1STR	0x03F25	R	Serial interface 1 status register	XIII-26	√	√	√	√
	RXBUF1	0x03F26	R	Serial interface 1 reception data buffer	XIII-21	√	√	√	√
	TXBUF1	0x03F27	R/W	Serial interface 1 transmission data buffer	XIII-21	√	√	√	√
	P5ODC	0x03EF5	R/W	Port 5 Nch open-drain control register	V-48	√	√	√	√
	P5DIR	0x03E95	R/W	Port 5 direction control register	V-46	√	√	√	√
	P5PLU(D)	0x03EA5	R/W	Port 5 pull-up resistor control register	V-47	√	√	√	√
	P7ODC	0x03EF7	R/W	Port 7 Nch open-drain control register	V-80	√	√	√	√
	P7DIR	0x03E97	R/W	Port 7 direction control register	V-79	√	√	√	√
	P7PLU	0x03EA7	R/W	Port 7 pull-up resistor control register	V-80	√	√	√	√
	PERIILR	0x03FFE	R/W	Peripheral function group interrupt control register	IV-35	√	√	√	√
	IRQEXPEN	0x03F4E	R/W	Peripheral function group input enable register	IV-36	√	√	√	√
	IRQEXPDT	0x03F4F	R/W	Peripheral function group interrupt factor retention register	IV-37	√	√	√	√
	SCINTSEL	0x03F28	R/W	Clock synchronous serial reception interrupt control register	XIII-27	√	√	√	√

R/W: Readable/Writable
R: Read only

Table remarks √: With function -: Without function

	Register	Address	R/W	Function	Page	MN101 EFA8	MN101 EFA3	MN101 EFA7	MN101 EFA2
Serial 2	SC2SEL	0x03F30	R/W	Serial 2 I/O pin switching control register	XIII-18	√	√	√	√
	SC2MD0	0x03F31	R/W	Serial interface 2 mode register 0	XIII-22	√	√	√	√
	SC2MD1	0x03F32	R/W	Serial interface 2 mode register 1	XIII-23	√	√	√	√
	SC2MD2	0x03F33	R/W	Serial interface 2 mode register 2	XIII-24	√	√	√	√
	SC2MD3	0x03F34	R/W	Serial interface 2 mode register 3	XIII-25	√	√	√	√
	SC2STR	0x03F35	R	Serial interface 2 status register	XIII-26	√	√	√	√
	RXBUF2	0x03F36	R	Serial interface 2 reception data buffer	XIII-21	√	√	√	√
	TXBUF2	0x03F37	R/W	Serial interface 2 transmission data buffer	XIII-21	√	√	√	√
	P6ODC	0x03EF6	R/W	Port 6 Nch open-drain control register	V-72	√	√	√	√
	P6DIR	0x03E96	R/W	Port 6 direction control register	V-70	√	√	√	√
	P6PLU	0x03EA6	R/W	Port 6 pull-up resistor control register	V-71	√	√	√	√
	PERIILR	0x03FFE	R/W	Peripheral function group interrupt control register	IV-35	√	√	√	√
	IRQEXPEN	0x03F4E	R/W	Peripheral function group input enable register	IV-36	√	√	√	√
	IRQEXPDT	0x03F4F	R/W	Peripheral function group interrupt factor retention register	IV-37	√	√	√	√
	SCINTSEL	0x03F28	R/W	Clock synchronous serial reception interrupt control register	XIII-27	√	√	√	√

R/W: Readable/Writable
R: Read only

Table remarks √: With function -: Without function

	Register	Address	R/W	Function	Page	MN101 EFA8	MN101 EFA3	MN101 EFA7	MN101 EFA2
Serial 4	SC4MD0	0x03F50	R/W	Serial interface 4 mode register 0	XIII-22	√	√	√	√
	SC4MD1	0x03F51	R/W	Serial interface 4 mode register 1	XIII-23	√	√	√	√
	SC4MD2	0x03F52	R/W	Serial interface 4 mode register 2	XIII-24	√	√	√	√
	SC4MD3	0x03F53	R/W	Serial interface 4 mode register 3	XIII-25	√	√	√	√
	SC4AD0	0x03F54	R/W	Serial interface 4 address setting register 0	XIII-33	√	√	√	√
	SC4AD1	0x03F55	R/W	Serial interface 4 address setting register 1	XIII-33	√	√	√	√
	SC4STR0	0x03F56	R/W	Serial interface 4 status register 0	XIII-34	√	√	√	√
	SC4STR1	0x03F57	R/W	Serial interface 4 status register 1	XIII-35	√	√	√	√
	RXBUF4	0x03F58	R	Serial interface 4 reception data buffer	XIII-28	√	√	√	√
	TXBUF4	0x03F59	R/W	Serial interface 4 transmission data buffer	XIII-28	√	√	√	√
	SC4SEL	0x03F5A	R/W	Serial 4 I/O pin switching control register	XIII-19	√	√	√	√
	P3ODC	0x03EF3	R/W	Port 3 Nch open-drain control register	V-28	√	√	-	-
	P3DIR	0x03E93	R/W	Port 3 direction control register	V-27	√	√	-	-
	P3PLUD	0x03EA3	R/W	Port 3 pull-up/pull-down resistor control register	V-27	√	√	-	-
	P7ODC	0x03EF7	R/W	Port 7 Nch open-drain control register	V-80	√	√	√	√
	P7DIR	0x03E97	R/W	Port 7 direction control register	V-79	√	√	√	√
	P7PLU	0x03EA7	R/W	Port 7 pull-up resistor control register	V-80	√	√	√	√
	IRQEXPEN	0x03F4E	R/W	Peripheral function group input enable register	IV-36	√	√	√	√
	IRQEXPDT	0x03F4F	R/W	Peripheral function group interrupt factor retention register	IV-37	√	√	√	√
	SCINTSEL	0x03F28	R/W	Clock synchronous serial reception interrupt control register	XIII-27	√	√	√	√

R/W: Readable/Writable
R: Read only

Set forced reset for serial interface by setting both SC4SBIS and SC4SBOS flags of SC4MD1 to “0”, before changing setting values of Serial Interface 4 mode registers (except for SC4STE, SC4STPC, SC4REX and SC4ACK0 flags) and address setting registers.

13.2.2 Serial I/O Pin Switching Control Registers

Pins are shared between serial interfaces.

SCnSEL registers are used for switching pins.

■ Serial 0 I/O Pin Switching Control Register (SC0SEL: 0x03F10)

MN101EFA8/A3

bp	7	6	5	4	3	2	1	0
Flag	SBO0SEL	SC0BRP2	SC0BRP1	SC0BRP0	OSL0	SC0SEL2	SC0SEL1	SC0SEL0
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7	SBO0SEL	UART reverse output selection 0: UART output 1: UART reverse output
6 to 4	SC0BRP2 SC0BRP1 SC0BRP0	Timer clock output selection XX0: Timer output 001: Timer output/2 011: Timer output/4 101: Timer output/8 111: Timer output/16
3	OSL0	Serial output port selection 0: A type 1: B type
2 to 0	SC0SEL2 SC0SEL1 SC0SEL0	Timer selection X00: Timer 0 X01: Timer 1 010: Timer 2 011: Timer 3 110: Setting prohibited 111: Timer A

■ Serial 0 I/O Pin Switching Control Register (SC0SEL: 0x03F10)

MN101EFA7/A2

bp	7	6	5	4	3	2	1	0
Flag	SBO0SEL	SC0BRP2	SC0BRP1	SC0BRP0	Reserved	SC0SEL2	SC0SEL1	SC0SEL0
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7	SBO0SEL	UART reverse output selection 0: UART output 1: UART reverse output
6 to 4	SC0BRP2 SC0BRP1 SC0BRP0	Timer clock output selection XX0: Timer output 001: Timer output/2 011: Timer output/4 101: Timer output/8 111: Timer output/16
3	Reserved	Always set to “0“
2 to 0	SC0SEL2 SC0SEL1 SC0SEL0	Timer selection X00: Timer 0 X01: Timer 1 010: Timer 2 011: Timer 3 110: Setting prohibited 111: Timer A

■ Serial 1 I/O Pin Switching Control Register (SC1SEL: 0x03F20)

bp	7	6	5	4	3	2	1	0
Flag	SBO1SEL	SC1BRP2	SC1BRP1	SC1BRP0	OSL1	SC1SEL2	SC1SEL1	SC1SEL0
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7	SBO1SEL	UART reverse output selection 0: UART output 1: UART reverse output
6 to 4	SC1BRP2 SC1BRP1 SC1BRP0	Timer clock output selection XX0: Timer output 001: Timer output/2 011: Timer output/4 101: Timer output/8 111: Timer output/16
3	OSL1	Serial output port selection 0: A type 1: B type
2 to 0	SC1SEL2 SC1SEL1 SC1SEL0	Timer selection X00: Timer 0 X01: Timer 1 010: Timer 2 011: Timer 3 110: Setting prohibited 111: Timer A

■ Serial 2 I/O Pin Switching Control Register (SC2SEL: 0x03F30)

bp	7	6	5	4	3	2	1	0
Flag	SBO2SEL	SC2BRP2	SC2BRP1	SC2BRP0	Reserved	SC2SEL2	SC2SEL1	SC2SEL0
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7	SBO2SEL	UART reverse output selection 0: UART output 1: UART reverse output
6 to 4	SC2BRP2 SC2BRP1 SC2BRP0	Timer clock output selection XX0: Timer output 001: Timer output/2 011: Timer output/4 101: Timer output/8 111: Timer output/16
3	Reserved	Always set to “0”
2 to 0	SC2SEL2 SC2SEL1 SC2SEL0	Timer selection X00: Timer 0 X01: Timer 1 010: Timer 2 011: Timer 3 110: Setting prohibited 111: Timer A

- Serial 4 I/O Pin Switching Control Register (SC4SEL: 0x03F5A)

MN101EFA8/A3

bp	7	6	5	4	3	2	1	0
Flag	Reserved	SC4BRP2	SC4BRP1	SC4BRP0	OSL2	SC4SEL2	SC4SEL1	SC4SEL0
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7	Reserved	Always set to “0”.
6 to 4	SC4BRP2 SC4BRP1 SC4BRP0	Timer clock output selection XX0: Timer output 001: Timer output/2 011: Timer output/4 101: Timer output/8 111: Timer output/16
3	OSL2	Serial output port selection 0: A type 1: B type
2 to 0	SC4SEL2 SC4SEL1 SC4SEL0	Timer selection X00: Timer 0 X01: Timer 1 010: Timer 2 011: Timer 3 110: Setting prohibited 111: Timer A

■ Serial 4 I/O Pin Switching Control Register (SC4SEL: 0x03F5A)

MN101EFA7/A2

bp	7	6	5	4	3	2	1	0
Flag	Reserved	SC4BRP2	SC4BRP1	SC4BRP0	Reserved	SC4SEL2	SC4SEL1	SC4SEL0
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7	Reserved	Always set to “0”.
6 to 4	SC4BRP2 SC4BRP1 SC4BRP0	Timer clock output selection XX0: Timer output 001: Timer output/2 011: Timer output/4 101: Timer output/8 111: Timer output/16
3	Reserved	Always set to “0”.
2 to 0	SC4SEL2 SC4SEL1 SC4SEL0	Timer selection X00: Timer 0 X01: Timer 1 010: Timer 2 011: Timer 3 110: Setting prohibited 111: Timer A

13.2.3 Serial Interface 0, 1 and 2 Control Registers

Serial interfaces 0, 1 and 2 can be used for clock synchronous and UART (duplex) communication.

Each serial interface is composed of 2 buffers and 6 registers.

- Serial interface n reception data buffer (RXBUFn)
- Serial interface n transmission data buffer (TXBUFn)
- Serial interface n mode register 0 (SCnMD0)
- Serial interface n mode register 1 (SCnMD1)
- Serial interface n mode register 2 (SCnMD2)
- Serial interface n mode register 3 (SCnMD3)
- Serial interface n status register (SCnSTR)

"n"= 0, 1 and 2 for Serial Interface 0, 1 and 2 respectively in [Chapter 13 13.2.3 Serial Interface 0, 1 and 2 Control Registers].

■ Serial Interface n Reception Data Buffer
(RXBUF0: 0x03F16, RXBUF1: 0x03F26, RXBUF2: 0x03F36)

bp	7	6	5	4	3	2	1	0
Flag	RXBUFn7	RXBUFn6	RXBUFn5	RXBUFn4	RXBUFn3	RXBUFn2	RXBUFn1	RXBUFn0
At reset	X	X	X	X	X	X	X	X
Access	R	R	R	R	R	R	R	R

■ Serial Interface n Transmission Data Buffer
(TXBUF0: 0x03F17, TXBUF1: 0x03F27, TXBUF2: 0x03F37)

bp	7	6	5	4	3	2	1	0
Flag	TXBUFn7	TXBUFn6	TXBUFn5	TXBUFn4	TXBUFn3	TXBUFn2	TXBUFn1	TXBUFn0
At reset	X	X	X	X	X	X	X	X
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- ■ Serial Interface n Mode Register 0 (SC0MD0: 0x03F11, SC1MD0: 0x03F21, SC2MD0: 0x03F31)

bp	7	6	5	4	3	2	1	0
Flag	SCnCE1	SCnSSC	SCnCTM	SCnDIR	SCnSTE	SCnLNG2	SCnLNG1	SCnLNG0
At reset	0	0	0	0	0	1	1	1
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description	
7	SCnCE1	Transmission data output edge 0: Falling 1: Rising	Reception data input edge 0: Rising 1: Falling
6	SCnSSC	Internal operation clock source 0: Input through SBT 1: Internal generation clock	
5	SCnCTM	Continuous transmission mode 0: Transfer with the communication blank mode 1: Transfer without the communication blank mode	
4	SCnDIR	Transfer bit specification 0: MSB first 1: LSB first	
3	SCnSTE	Start condition selection 0: Disable 1: Enable	
2 to 0	SCnLNG2 SCnLNG1 SCnLNG0	Synchronous serial transfer bit count 000: Setting prohibited 001: 2 bits 010: 3 bits 011: 4 bits 100: 5 bits 101: 6 bits 110: 7 bits 111: 8 bits	

To change setting values of mode registers, set forced reset first for serial interface by setting both SCnSBIS and SCnSBOS flags of SCnMD1 to “0”. (n = 0 to 2)

■ Serial Interface n Mode Register 1 (SC0MD1: 0x03F12, SC1MD1: 0x03F22, SC2MD1: 0x03F32)

bp	7	6	5	4	3	2	1	0
Flag	SCnIOM	SCnSBTS	SCnSBIS	SCnSBOS	SCnCKM	SCnMST	SCnDIV	SCnCMD
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

<table>
<tr><th>bp</th><th>Flag</th><th colspan="2">Description</th></tr>
<tr><td>7</td><td>SCnIOM</td><td colspan="2">Serial data input pin selection
0: Data input from SBIn (RXDn)
1: Data input from SBOn (TXDn)</td></tr>
<tr><td>6</td><td>SCnSBTS</td><td colspan="2">SBT pin function selection
0: Port
1: Transfer clock I/O</td></tr>
<tr><td>5</td><td>SCnSBIS</td><td colspan="2">Serial input control selection
0: "1" input
1: Serial data input</td></tr>
<tr><td>4</td><td>SCnSBOS</td><td colspan="2">SBO (TXD) pin function selection
0: Port
1: Serial data output</td></tr>
<tr><td>3</td><td>SCnCKM</td><td>Clock synchronous (SCnCMD=0)
Transfer clock division selection
0: Not divided
1: Divided</td><td>UART (SCnCMD=1)
Transfer clock is divided regardless of the setting value.
The data read out is the setting value.</td></tr>
<tr><td>2</td><td>SCnMST</td><td colspan="2">Clock master/salve selection
0: Clock slave
1: Clock master</td></tr>
<tr><td>1</td><td>SCnDIV</td><td colspan="2">Transfer clock division value selection
0: Divide by 8
1: Divide by 16</td></tr>
<tr><td>0</td><td>SCnCMD</td><td colspan="2">Clock synchronous/UART selection
0: Clock synchronous
1: UART</td></tr>
</table>

To set serial interface communication mode to “UART”, set Serial Interface n mode register 1 (SCnMD1) while the serial data input pin is at “High”.

To change setting values of mode registers, set forced reset for serial interface at first by setting both SCnSBIS and SCnSBOS flags of SCnMD1 to “0” (n=0 to 2).

If setting both the SCnSBIS flag and the SCnSBOS flag to "0", the serial interface functions are forced to reset and stopped.

■ Serial Interface n Mode Register 2 (SC0MD2: 0x03F13, SC1MD2: 0x03F23, SC2MD2: 0x03F33)

bp	7	6	5	4	3	2	1	0
Flag	SCnFM1	SCnFM0	SCnPM1	SCnPM0	SCnNPE	-	SCnBRKF	SCnBRKE
At reset	0	0	0	0	0	-	0	0
Access	R/W	R/W	R/W	R/W	R/W	-	R	R/W

<table>
<tr><th>bp</th><th>Flag</th><th colspan="2">Description</th></tr>
<tr><td>7-6</td><td>SCnFM1
SCnFM0</td><td colspan="2">Frame mode specification
00: Data 7 bits + stop 1 bit
01: Data 7 bits + stop 2 bits
10: Data 8 bits + stop 1 bit
11: Data 8 bits + stop 2 bits</td></tr>
<tr><td rowspan="2">5-4</td><td rowspan="2">SCnPM1
SCnPM0</td><td colspan="2">Additional bit specification</td></tr>
<tr><td>At transmission
00: Add "0"
01: Add "1"
10: Add odd parity
11: Add even parity</td><td>At reception
00: Check for "0"
01: Check for "1"
10: Check for odd parity
11: Check for even parity</td></tr>
<tr><td>3</td><td>SCnNPE</td><td colspan="2">Parity enable
0: Enable parity bit
1: Disable parity bit</td></tr>
<tr><td>2</td><td>-</td><td colspan="2">-</td></tr>
<tr><td>1</td><td>SCnBRKF</td><td colspan="2">Break status receive monitor
0: Data reception
1: Break reception</td></tr>
<tr><td>0</td><td>SCnBRKE</td><td colspan="2">Break status transmit control
0: Data transmission
1: Break transmission</td></tr>
</table>

■ Serial Interface n Mode Register 3 (SC0MD3: 0x03F14, SC1MD3: 0x03F24, SC2MD3: 0x03F34)

bp	7	6	5	4	3	2	1	0
Flag	SCnFDC1	SCnFDC0	-	-	SCnPSCE	SCnPSC2	SCnPSC1	SCnPSC0
At reset	0	0	-	-	0	0	0	0
Access	R/W	R/W	-	-	R/W	R/W	R/W	R/W

bp	Flag	Description
7-6	SCnFDC1 SCnFDC0	Output selection after SBO final data transmit 00: Fixed at "High" output 01: Hold the final data 10: Fixed at "Low" output 11: Setting prohibited
5-4	-	-
3	SCnPSCE	Prescaler count control 0: Disable count 1: Enable count
2 to 0	SCnPSC2 SCnPSC1 SCnPSC0	Clock selection 000: fpll-div/2 001: fpll-div/4 010: fpll-div/16 011: fpll-div/64 100: fs/2 101: fs/4 11X: Timer output * Timer 0 to 3 or A can be selected by the SCnSEL2 to 0 flags of SCnSEL register.

- Serial Interface n Status Register (SC0STR: 0x03F15, SC1STR: 0x03F25, SC2STR: 0x03F35)

bp	7	6	5	4	3	2	1	0
Flag	SCnTBSY	SCnRBSY	SCnTEMP	SCnREMP	SCnFEF	SCnPEK	SCnORE	SCnERE
At reset	0	0	0	0	0	0	0	0
Access	R	R	R	R	R	R	R	R

bp	Flag	Description
7	SCnTBSY	Serial bus status 0: Serial transmission NOT in progress 1: Serial transmission in progress
6	SCnRBSY	Serial bus status 0: Serial reception NOT in progress 1: Serial reception in progress
5	SCnTEMP	Transmission buffer empty flag 0: Empty 1: Full
4	SCnREMP	Reception buffer empty flag 0: Empty 1: Full
3	SCnFEF	Framing error detection 0: No error 1: Error
2	SCnPEK	Parity error detection 0: No error 1: Error
1	SCnORE	Overrun error detection 0: No error 1: Error
0	SCnERE	Error monitor flag 0: No error 1: Error

■ Clock Synchronous Serial Reception Interrupt Control Register (SCINTSEL: 0x03F28)

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	SC4IGC	SC2IGC	SC1IGC	SC0IGC
At reset	-	-	-	-	0	0	0	0
Access	-	-	-	-	R/W	R/W	R/W	R/W

bp	Flag	Description
7-4	-	-
3	SC4IGC	SC4STPCIRQ generation timing at clock synchronous serial data reception 0: Not used 1: SBT4 falling
2	SC2IGC	SC2RIRQ generation timing at clock synchronous serial data reception 0: Reception is completed 1: SBT2 falling
1	SC1IGC	SC1RIRQ generation timing at clock synchronous serial data reception 0: Reception is completed 1: SBT1 falling
0	SC0IGC	SC0RIRQ generation timing at clock synchronous serial data reception 0: Reception is completed 1: SBT0 falling

The SCnIGC flag of SCINTSEL register can only be set to "1" during clock synchronous slave reception.
If other communication setting is used, the SCnIGC flag must be set to "0".

13.2.4 Serial Interface 4 Control Registers

Serial Interface 4 can be used for clock synchronous and multi master IIC communication.

This serial interface is composed of 2 buffers and 8 registers.

- Serial Interface 4 reception data buffer (RXBUF4)
- Serial Interface 4 transmission data buffer (TXBUF4)
- Serial Interface 4 mode register 0 (SC4MD0)
- Serial Interface 4 mode register 1 (SC4MD1)
- Serial Interface 4 mode register 2 (SC4MD2)
- Serial Interface 4 mode register 3 (SC4MD3)
- Serial Interface 4 address set register 0 (SC4AD0)
- Serial Interface 4 address set register 1 (SC4AD1)
- Serial Interface 4 status register 0 (SC4STR0)
- Serial Interface 4 status register 1 (SC4STR1)

■ Serial Interface 4 Reception Data Buffer (RXBUF4: 0x03F58)

bp	7	6	5	4	3	2	1	0
Flag	RXBUF47	RXBUF46	RXBUF45	RXBUF44	RXBUF43	RXBUF42	RXBUF41	RXBUF40
At reset	X	X	X	X	X	X	X	X
Access	R	R	R	R	R	R	R	R

■ Serial Interface 4 Transmission Data Buffer (TXBUF4: 0x03F59)

bp	7	6	5	4	3	2	1	0
Flag	TXBUF47	TXBUF46	TXBUF45	TXBUF44	TXBUF43	TXBUF42	TXBUF41	TXBUF40
At reset	X	X	X	X	X	X	X	X
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

13.2.5 Serial Interface 4 Mode Register

- Serial Interface 4 Mode Register 0 (SC4MD0: 0x03F50)

bp	7	6	5	4	3	2	1	0
Flag	SC4CE1	SC4SSC	SC4DEM	SC4DIR	SC4STE	SC4LNG2	SC4LNG1	SC4LNG0
At reset	0	0	0	0	0	1	1	1
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description	
7	SC4CE1	Transmission data output edge 0: Falling 1: Rising	Reception data input edge 0: Rising 1: Falling
6	SC4SSC	Internal operation clock source 0: Input through SBT 1: Internal generation clock	
5	SC4DEM	Operation selection after detecting communication data instability in IIC slave communication mode 0: Start condition is not detected 1: Start condition is detected	
4	SC4DIR	Transfer bit specification 0: MSB first 1: LSB first	
3	SC4STE	Start condition selection 0: Disable 1: Enable	
2 to 0	SC4LNG2 SC4LNG1 SC4LNG0	Transfer bit count 000: Setting prohibited 001: 2 bits 010: 3 bits 011: 4 bits 100: 5 bits 101: 6 bits 110: 7 bits 111: 8 bits	

Set forced reset for serial interface by setting both SC4SBIS and SC4SBOS flags of SC4MD1 to “0”, before changing setting values of Serial Interface 4 mode registers (except for SC4STE, SC4STPC, SC4REX and SC4ACK0 flags) and address setting registers.

■ Serial Interface 4 Mode Register 1 (SC4MD1: 0x03F51)

bp	7	6	5	4	3	2	1	0
Flag	SC4IOM	SC4SBTS	SC4SBIS	SC4SBOS	-	SC4MST	-	SC4CTM
At reset	0	0	0	0	-	0	-	0
Access	R/W	R/W	R/W	R/W	-	R/W	-	R/W

bp	Flag	Description
7	SC4IOM	Serial data input pin selection 0: Data input form SBI4 1: Data input from SBO4 (SDA4)
6	SC4SBTS	SBT4 pin function selection 0: Port 1: Transfer clock I/O
5	SC4SBIS	Serial input control selection 0: "1" input 1: Serial input
4	SC4SBOS	SBO4 (SDA4) pin function selection 0: Port 1: Serial data output
3	-	-
2	SC4MST	Clock master/Clock salve selection 0: Clock slave 1: SC4CMD=0: Clock master SC4CMD=1: Clock master/Clock slave
1	-	-
0	SC4CTM	Continuous transmission mode 0: Transfer with the communication blank mode 1: Transfer without the communication blank mode

Set forced reset for serial interface by setting both SC4SBIS and SC4SBOS flags of SC4MD1 to “0”, before changing setting values of Serial Interface 4 mode registers (except for SC4STE, SC4STPC, SC4REX and SC4ACK0 flags) and address setting registers.

■ Serial Interface 4 Mode Register 2 (SC4MD2: 0x03F52)

bp	7	6	5	4	3	2	1	0
Flag	SC4FDC1	SC4FDC0	-	-	SC4PSCE	SC4PSC2	SC4PSC1	SC4PSC0
At reset	0	0	-	-	0	0	0	0
Access	R/W	R/W	-	-	R/W	R/W	R/W	R/W

bp	Flag	Description
7 to 6	SC4FDC1 SC4FDC0	Output selection after SBO final data transmit 00: Fixed to "High" output 01: Fixed to "Low" output 10: Hold the final data 11: Setting prohibited
5 to 4	-	-
3	SC4PSCE	Prescaler count control 0: Disable count 1: Enable count
2 to 0	SC4PSC2 SC4PSC1 SC4PSC0	Clock selection 000: fpll-div/2 001: fpll-div/4 010: fpll-div/16 011: fpll-div/32 100: fs/2 101: fs/4 11X: Timer output * Timer 0 to 3 or A can be selected by the SC4SEL2 to 0 flags of SC4SEL register.

■ Serial Interface 4 Mode Register 3 (SC4MD3: 0x03F53)

bp	7	6	5	4	3	2	1	0
Flag	Reserved	SC4ADM	SC4STPC	SC4TMD	SC4REX	SC4CMD	SC4ACKS	SC4ACKO
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description	
7	Reserved	Always set to “0”.	
6	SC4ADM	Address format selection 0: 7-bit address 1: 10-bit address	
5	SC4STPC	Stop condition generation request flag in IIC communication 0: Disable 1: Stop condition is generated	
4	SC4TMD	Communication mode selection in IIC communication 0: Standard mode 1: High-speed mode	
3	SC4REX	Transmission/reception mode selection in IIC communication 0: Transmission 1: Reception	
2	SC4CMD	Clock synchronous/IIC selection 0: Clock synchronous 1: IIC	
1	SC4ACKS	ACK bit enable 0: Disable 1: Enable	
0	SC4ACKO	At transmission ACK/NACK bit detection flag 0: ACK detection 1: NACK detection	At reception ACK/NACK bit transmission flag 0: ACK transmission 1: NACK transmission

SC4STPC and SC4ACKO flags are updated by hardware. Thus, when values of these flags are read out, the values might be different from the values which are previously written.
When the SC4STPC flag is set to “1”, the circuit will hold the status (flag:1) for a certain period of time. And then, the flag is cleared to “0”.
The written value of the SC4ACKO flag will be reloaded when the communication starts.
With the features above, when changing SC4MD3 register, set the value of the register with careful attention to settings of the SC4STPC flag and the SC4ACKO flag.
In particular, do no execute Read-Modify-Write operation, including bit manipulation instructions such as BSET/BCLR.
Otherwise, multiple accesses may rewrite the value of the SC4MD3 register to an unexpected value.

Set forced reset for serial interface by setting both SC4SBIS and SC4SBOS flags of SC4MD1 to “0”, before changing setting values of Serial Interface 4 mode registers (except for SC4STE, SC4STPC, SC4REX and SC4ACK0 flags) and address setting registers.

13.2.6 Serial Interface 4 Address Set Register

Serial Interface 4 has address setting registers for 10-bit data.

- Serial Interface 4 Address Set Register 0 (SC4AD0: 0x03F54)

bp	7	6	5	4	3	2	1	0
Flag	SC4ADR7	SC4ADR6	SC4ADR5	SC4ADR4	SC4ADR3	SC4ADR2	SC4ADR1	SC4ADR0
At reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- Serial Interface 4 Address Set Register 1 (SC4AD1: 0x03F55)

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	-	-	SC4ADR9	SC4ADR8
At reset	-	-	-	-	-	-	0	0
Access	-	-	-	-	-	-	R/W	R/W

13.2.7 Serial Interface 4 Status Register

■ Serial Interface 4 Status Register 0 (SC4STR0: 0x03F56)

bp	7	6	5	4	3	2	1	0
Flag	SC4BSY	-	SC4TEMP	SC4REMP	-	-	-	SC4ORE
At reset	0	-	0	0	-	-	-	0
Access	R	-	R	R	-	-	-	R/W

bp	Flag	Description
7	SC4BSY	Clock synchronous communication serial bus status 0: Serial transmission NOT in progress 1: Serial transmission in progress
6	-	-
5	SC4TEMP	Transmission buffer empty flag 0: Empty 1: Full
4	SC4REMP	Reception buffer empty flag 0: Empty 1: Full
3 to 1	-	-
0	SC4ORE	Overrun error detection 0: No error 1: Error

Regarding SC4ORE flag:
when “1” is written, the value of this flag does not change, and
when “0” is written, the value of this flag is cleared to “0”.

- Serial Interface 4 Status Register 1 (SC4STR1: 0x03F57)

bp	7	6	5	4	3	2	1	0
Flag	SC4 WRS	SC4 ABT_LST	SC4 ADD_ACC	SC4 STRT	SC4 BUSBSY	SC4 IICBSY	SC4 GCALL	SC4 DATA_ERR
At reset	0	0	0	0	0	0	0	0
Access	R	R/W	R	R	R	R	R	R/W

bp	Flag	Description
7	SC4WRS	Data transfer determination flag in slave communication 0: Master to Slave 1: Slave to Master
6	SC4ABT_LST	Arbitration lost detection flag 0: Not detected 1: Detected
5	SC4ADD_ACC	Slave address comparison flag 0: Address is unmatched 1: Address is matched
4	SC4STRT	Start condition detection flag 0: Not detected 1: Detected
3	SC4BUSBSY	Bus busy flag 0: Bus free status 1: Bus busy status
2	SC4IICBSY	Serial bus status in IIC communication 0: Serial transmission NOT in progress 1: Serial transmission in progress
1	SC4GCALL	General call detection flag 0: Not detected 1: Detected
0	SC4DATA_ERR	Communication error detection flag 0: Not detected 1: Detected

Regarding SC4ABT_LST flag and SC4DATA_ERR flag:
when “1” is written, the values of these flags do not change, and
when “0” is written, the values of these flags are cleared to “0”.

13.2.8 Use Flags of Each Function

Table:13.2.2 shows the necessary flags for settings of clock synchronous, duplex UART and multi master IIC.

Table:13.2.2 List of used flags

√: Setting necessity -: Setting unnecessity

		Clock Synchronous	Duplex UART	Multi master IIC
SCnMD0 (n = 0 to 2)	SCnCE1	√	-	-
	SCnSSC	√	-	
	SCnCTM	√	-	
	SCnDIR	√	√	
	SCnSTE	√	-	
	SCnLNG2-0	√	-	
SCnMD1 (n = 0 to 2)	SCnIOM	√	√	-
	SCnSBTS	√	√	
	SCnSBIS	√	√	
	SCnSBOS	√	√	
	SCnCKM	√	√	
	SCnMST	√	-	
	SCnDIV	√	-	
	SCnCMD	√	√	
SCnMD2 (n= 0 to 2)	SCnFM1-0	-	√	-
	SCnPM1-0	-	√	
	SCnNPE	-	√	
	SCnBRKE	-	√	
SCnMD3 (n = 0 to 2)	SCnFDC1-0	√	-	-
	SCnPSCE	√	√	
	SCnPSC2-0	√	√	
SC4MD0	SC4CE1	√	-	-
	SC4SSC	√		-
	SC4DEM	-		√
	SC4DIR	√		√
	SC4STE	√		√
	SC4LNG2-0	√		√
SC4MD1	SC4IOM	√	-	√
	SC4SBTS	√		√
	SC4SBIS	√		√
	SC4SBOS	√		√
	SC4MST	√		√
	SC4CTM	√		-
SC4MD2	SC4FDC1-0	√	-	-
	SC4PSCE	√		√
	SC4PSC2-0	√		√
SC4MD3	SC4ADM	-	-	√
	SC4STPC	-		√
	SC4TMD	-		√
	SC4REX	-		√
	SC4CMD	√		√
	SC4ACKS	-		√
	SC4ACKO	-		√

13.3 Clock Synchronous Serial Interface

13.3.1 Operation

This section describes the clock synchronous communication method of serial interfaces 0, 1, 2 and 4.

"n"= 0, 1, 2 and 4 for Serial Interface 0, 1, 2 and 4 respectively in [Chapter 13 13.3 Clock Synchronous Serial Interface].

■ Communication Type

The communication mode can be selected from 3-wire type (clock pin (SBTn pin), data output pin (SBOn pin), data input pin (SBIn pin)) or 2-wire type (clock pin (SBTn pin), data I/O pin (SBOn pin)). Set the communication mode by the SCnIOM flag of the SCnMD1 register. In 2-wire reception, select "serial data input" by setting the SCnSBIS flag of the SCnMD1 register to "1". The SBIn pin can be used as a general-purpose port.

■ Activation Factor for Communication

Table:13.3.1 shows activation factors for communication. In master communication, the transfer clock is generated by communication activation factors. In slave communication, except during communication, the signal input from SBTn pin is masked in serial interface to prevent errors by noise or so; thus, input an external clock after releasing the mask by communication activation factors.

In addition, if "set transmission data" or "set dummy data" is performed as a communication activation factor, input the external clock after 3.5 transfer clock interval or longer after the data is set to TXBUFn. This wait time is needed to load the data from TXBUFn to internal shift register.

Table:13.3.1 Synchronous Serial Interface Activation Factors

	Activation factors	
	Transmission	Reception
Master communication	Set transmission data	Set dummy data (Without start condition)
		Input start condition (With start condition)
Slave communication	Input clock after transmission data is set	Input clock after dummy data is set (Without start condition)
		Input clock after start condition is input (With start condition)

SBTn pin is masked in serial interface to prevent errors caused by noise except during communication. In slave communication, set data to TXBUFn or input a start condition before inputting clock to SBTn pin.

Wait 3.5 transfer clocks or longer before inputting the external clock after the data set to TXBUFn. Otherwise, proper operation cannot be guaranteed.

In the reception with a start condition, writing any data into the transmission data buffer is prohibited. Writing to this buffer causes improper operation of the serial interface.

■ Transfer Bit Setup

The transfer bit count can be selected from 2 to 8 bits. Set the SCnLNG2 to 0 flag of SCnMD0 register (at reset: 111). The SCnLNG2 to 0 flags hold the previous set value until other value is set.

■ Start Condition Setup

The SCnSTE flag of SCnMD0 register sets whether a start condition is enabled or disabled. When the data (SBIn (3-wire) or SBOn (2-wire)) pin changes from "High" to "Low" during the clock (SBTn) pin = "High", setting the SCnCE1 flag of SCnMD0 register to "0" allows a start condition to be recognized. When the data (SBIn (3-wire) or SBOn pin (2-wire)) pin changes from "High" to "Low" during the clock (SBTn) pin = "Low", setting the SCnCE1 flag to "1" allows a start condition to be recognized. Set the SCnSBOS flag and the SCnSBIS flag of SCnMD1 register to "0" respectively to change the start condition enable/disable setting. When operating transmission and reception at the same time, select "start condition disable"; otherwise, proper operation cannot be guaranteed.

■ First Transfer Bit Setup

Either MSB first or LSB first at transfer can be selected. Set the first bit by the SCnDIR flag of SCnMD0 register.

■ Transmission Data Buffer

The transmission data buffer TXBUFn is a spare buffer which stores data to be loaded into internal shift register. Store the data to be transmitted into transmission data buffer TXBUFn. The data is automatically loaded into internal shift register. The data loading period of 3 transfer clocks is needed for loading data. During the data loading period, setting another data in TXBUFn may cause an error. The transmission buffer empty flag SCnTEMP of SCnSTR register can be monitored to determined whether it is within a data loading period. When data is set in TXBUFn, the SCnTEMP flag is set to "1" until the data loading is finished. When communication is restarted, the SCnTEMP flag is automatically cleared to "0".

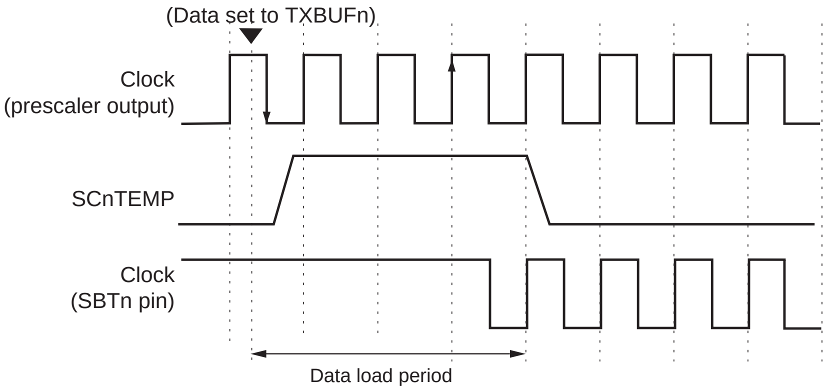


Figure:13.3.1 Transmission Data Buffer

■ Reception Data Buffer

The reception data buffer RXBUFn are the spare buffers which save the data received by reception shift register. After Serial Interface n interrupt SCn(T)IRQ is generated, data in internal shift register are automatically stored in the reception data buffers RXBUFn. RXBUFn can store data up to 1 byte. Read the data in RXBUFn before the next reception is completed since RXBUFn is rewritten every time the reception is completed. The reception data buffer empty flags SCnREMP is set to "1" after SCn(T)IRQ is generated. SCnREMP is cleared to "0" when reading RXBUFn.

When the start condition is received during the communication, the data reception is executed from the start again.
In this case, the data reception may not be completed normally.

RXBUFn is rewritten every time the reception is completed. In continuous communication, read the data in RXBUFn before the next reception is completed.

■ Transmit Bit Count and First Transfer Bit

In transmission, when the transfer bit is 2 bit to 7 bits, data storage method for the transmission data buffer differs depending on the first transfer bit specification. At MSB first, store data in the upper bits of TXBUFn. When there are 6 bits to be transferred, from "F" to "A", data "A" to "F" are stored in bp2 to bp7 of TXBUFn, as shown in Figure:13.3.2. At LSB first, store data in the lower bits of TXBUFn. When there are 6 bits to be transferred, from "F" to "A", data "A" to "F" are stored in bp0 to bp5 of TXBUFn, as shown in Figure:13.3.3.

	7	6	5	4	3	2	1	0
TXBUFn	F	E	D	C	B	A		

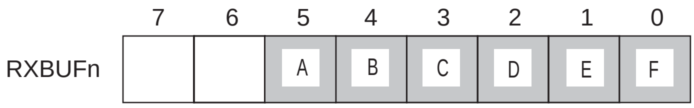
Figure:13.3.2 Transfer Bit Count and First Transfer Bit (At MSB First)

	7	6	5	4	3	2	1	0
TXBUFn			F	E	D	C	B	A

Figure:13.3.3 Transfer Bit Count and First Transfer Bit (At LSB First)

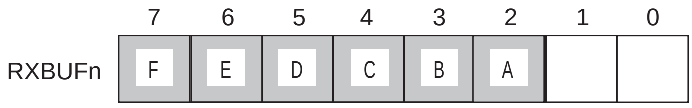
■ Reception Bit Count and First Transfer Bit

In reception, when the transfer bit is 2 to 7 bits, data storage method for the reception data buffer RXBUFn differs depending on the first transfer bit specification. At MSB first, reception data are stored in the lower bits of RXBUFn. When there are 6 bits to be transferred, with "A" to be the first bit to transfer and "F" to be the last bit to transfer, data "A" to "F" are stored in bp5 to bp0 of RXBUFn, as shown in Figure:13.3.4. At LSB first, data are stored in the upper bits of RXBUFn. When there are 6 bits to be transferred, with "A" to be the first bit to transfer and "F" to be the last bit to transfer, data "A" to "F" are stored in bp2 to bp7 of RXBUFn, as shown in Figure:13.3.5.



	7	6	5	4	3	2	1	0
RXBUFn			A	B	C	D	E	F

Figure:13.3.4 Reception Bit Count and Transfer First Bit (At MSB First)



	7	6	5	4	3	2	1	0
RXBUFn	F	E	D	C	B	A		

Figure:13.3.5 Reception Bit Count and Transfer First Bit (At LSB First)

When the reception transfer bit count shows 2 to 7 bits, data except the reception data of the specified transfer bit is undefined. The reception data should be masked to use with and instruction or others.

■ Continuous Communication

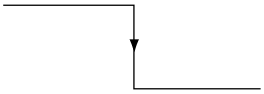
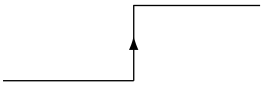
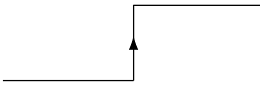
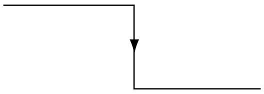
This serial interface has a continuous communication function. If data is set in the transmission data buffer TXBUFn during transmission, the transmission buffer empty flag SCnTEMP is set, and transmission is automatically continued. Set data in TXBUFn before Serial Interface n interrupt SCn(T)IRQ is generated after data is loaded into transmission shift register. Communication blanks, from SCn(T)IRQ generation to the next transfer clock output, are 4 transfer clocks.

When the slave reception is performed with the start condition "enable" in the continuous communication, the system configuration is needed to notify the master of the notification, the data which is read previously may be overwritten.

■ Input Edge/Output Edge Setup

An output edge of transmission data and an input edge of reception data can be set by the SCnCE1 flag of the SCnMD0 register. When the SCnCE1 flag = "0", transmission data is output synchronously at the falling edge of the clock; when the SCnCE1 flag = "1", the data is output synchronously at the rising edge. When the SCnCE1 flag = "0", reception data is loaded synchronously at the rising edge of the clock; when the SCnCE1 flag = "1", the data is loaded synchronously at the falling edge.

Table:13.3.2 Transmission Data Output Edge and Reception Data Input Edge

SCnCE1	Transmission data output edge	Reception data input edge
0		
1		

■ Clock Setup (Serial Interface 0, 1, 2 and 4)

The clock source can set with a dedicated prescaler or timer (2 systems) output. Set it by the SCnPSC2 to 0 of SCnMD3 register. The dedicated prescaler starts to operate when "prescaler operation" is selected by the SCnPSCE flag of SCnMD3 register. Either the internal clock (clock master) or the external clock (clock slave) can be selected by the SCnMST flag of SCnMD1 register. If selecting the external clock, set the internal clock that has the same clock cycle or lower to the external clock by SCnMD3 register. Table:13.3.3 shows the internal clock source that can be set by SCnMD3 register.

Table:13.3.3 Synchronous Serial Interface Clock Source

		Serial Interface 0	Serial Interface 1	Serial Interface 2	Serial Interface 4
Clock source (internal clock)	fpll-div/2	√	√	√	√
	fpll-div/4	√	√	√	√
	fpll-div/8	-	-	-	-
	fpll-div/16	√	√	√	√
	fpll-div/32	-	-	-	√
	fpll-div/64	√	√	√	-
	fs/2	√	√	√	√
	fs/4	√	√	√	√
	Timer 0 output	√	√	√	√
	Timer 1 output	√	√	√	√
	Timer 2 output	√	√	√	√
	Timer A output	√	√	√	√

■ Serial Interface Internal Operating Clock Source Selection

Serial interface internal clock source can be selected at clock synchronous master communication.
When the SCnSSC flag is set to "0", it is possible to communicate by operating the serial interface internal logic with the communication clock output from SBTn on the condition similar to the clock input to the slave device. When the SCnSSC flag is "1", the influence of the exogenous noise on the communication clock line can be decreased by operating the Serial Interface internal logic directly with the clock generated in the communication clock control circuit. The SCnSSC flag is not available on the clock synchronous slave communication, UART or multi master IIC mode.

The transfer speed should be up to 5.0 MHz. If the transfer clock exceeds 5.0 MHz, the transfer data may not be sent correctly.

Be sure to set the SCnSBIS flag and SCnSBOS flag of SCnMD1 register to "0" to switch clock setting.

■ Interrupt

When the clock synchronous serial interface is used, only one type interrupt, SCn(T)IRQ is available. SCn(T)IRQ is generated at the completion of transmission/reception.

■ Data Input Pin Setup

3-wire communication (clock pin (SBTn), data output pin (SBOn) and data input pin (SBIn)) or 2-wire type (clock pin (SBTn) and data I/O pin (SBOn)) can be selected as communication mode. Set it by the SCnIOM flag of SCnMDI register. SBIn pin can be used only for serial data input. SBOn can select to be a serial data input or output. Selecting "data input from SBOn" sets 2-wire communication; I/O mode selection of port direction control register controls direction of SBOn to switch between transmission/reception. At this time, SBIn can be used as a general-purpose port.

In reception, if SCnIOM of SCnMD1 register is set to "1" to select "serial data input from SBOn pin", SBIn pin can be used as a general-purpose port.

■ Reception Buffer Empty Flag

After completion of reception (Serial Interrupt n interrupt SCn(T)IRQ is generated), data is automatically stored into RXBUFn from internal shift register. If data is stored into shift register RXBUFn when the SCnSBIS flag of SC0MDn register is set to "serial input", the reception buffer empty flag SCnREMP of SCnSTR register is set to "1". This indicates that the received data is waiting to be read. SCnREMP is cleared to "0" by reading data of RXBUFn.

■ Transmission Buffer Empty Flag

During communication (until Serial Interrupt n interrupt SCn(T)IRQ is generated since data is loaded into internal shift register) if data is set in TXBUFn, the transmission buffer empty flag SCnTEMP of SCnSTR register is set to "1". This indicates that the next transmission is waiting to be loaded. When data is loaded to internal shift register from TXBUFn after SCn(T)IRQ is generated and SC0TEMP is cleared to "0", the next transfer starts automatically.

■ Overrun Error and Error Monitor Flag (Serial Interface 0, 1 and 2)

When the next data reception is completed before reading the data of reception data buffer RXBUFn received in previous communication, an overrun error is generated and the SCnORE flag of SCnSTR register is set to "1". At the same time, the error monitor flag SCnERE is set indicating that the reception has an error. The SCnORE flag is cleared when the next Serial Interrupt n interrupt SCnTIRQ is generated the data of RXBUFn is read. SCnERE is also cleared after the SCnORE flag is cleared. These error flags have no effects on the communication operation.

■ Reception BUSY Flag (Serial Interface 0, 1 and 2)

If data is set in RXBUFn or a start condition is recognized when the SCnSBIS flag of SCnMD1 register is set to "serial data input", the BUSY flag of SCnSTR register (SCnRBSY) is set to "1". The SCnRBSY flag is cleared to "0" after Serial Interface n interrupt SCnTIRQ is generated. During continuous communication, the set SCnRBSY flag is retained. If the transmission buffer empty flag SCnTEMP has already been cleared to "0" when Serial Interface n interrupt SCnTIRQ is generated, SCnRBSY is cleared to "0". If the SCnSBIS flag is set to "0" during communication, the SCnRBSY flag is cleared to "0".

■ Transmission BUSY Flag (Serial Interface 0, 1 and 2)

If data is set in TXBUFn or a start condition is recognized when the SCnSBOS flag of SCnMD1 register is set to "serial data output", the transmission Busy SCnTBSY flag of SCnSTR register is set. After Serial Interface n interrupt SCnTIRQ is generated, the flag is cleared to "0". During continuous communication, the set SCnTBSY flag is retained. If the transmission buffer empty flag SCnTEMP is cleared to "0" when Serial Interface n interrupt SCnTIRQ is generated, SCnTBSY flag has been already cleared to "0". If the SCnSBOS flag is set to "0" during communication, the SCnTBSY flag is reset to "0".

■ BUSY Flag (Serial Interface 4)

If the data is set in TXBUF4 or a start condition is recognized, SC4BSY flag of the SC4STR0 is set. It is cleared to "0" after Serial Interface 4 interrupt SC4IRQ is generated. During continuous communication, the set SC4BSY flag is retained. If the transmission buffer empty flag SC4TEMP has already been cleared to "0" when Serial Interface 4 interrupt SC4IRQ is generated, SC4BSY is cleared to "0".

■ Emergency Reset

This serial interface has an emergency reset function for malfunction. The SCnSBOS flag and the SCnSBIS flag of SCnMD1 register should be set to "0" (SBOn pin function: port, input data: "1" input) to operate an emergency reset.

At emergency reset, status register (all flags of SCnSTR, the SCnBRKF flag of SCnMD2 register in Serial Interface 0 and 1, all flags of SC4STR0 and the bp6 to bp0 flags of SC4STR1 registers and the SC4STPC flag of SC4MD3 register in Serial Interface 4) is initialized to the reset value, but the set value of the any other control registers are held.

■ Last Bit of Transmission Data

The following table shows the data output hold period of the last bit in transmission and the required minimum data input period of the last bit in reception. The internal clock should be set at slave to keep the data hold time in transmission.

Table:13.3.4 Last Bit Data Length of Transfer Data

	The last bit data hold period at transmission	The last bit data input period at reception
At master	1 bit data length	1 bit data length (minimum)
At slave	[1 bit data length of external clock × 1/2] + [internal clock cycle × (1/2 to 3/2)]	

When the start condition is disabled (SCnSTE flag = 0), SBOn output after the data output hold period of the last bit can be set as indicated in Table:13.3.5 by the set values of the SCnFDC1 to 0 flags of the SCnMD3 register.

After a reset is released, the output before a serial transfer is "High" despite the set value of the SCnFDC1 to 0 flags. When the start condition is enabled (SCnSTE flag = 1), "High" is output despite the set values of the SCnFDC1 to 0 flags.

Table:13.3.5 SBOn Output after Data Output Hold Period of Last Bit (Without Start Condition)

SCnFDC1 flag	SCnFDC0 flag	SBOn output after data output hold period of last bit
0	0	Fixed to "High" output
1	0	Fixed to "Low" output
0	1	Hold the final data
1	1	Setting prohibited

■ Other Control Flag Setup (Serial Interface 0, 1 and 2)

The flags shown below are not needed to be set or monitored since they are not used at clock synchronous communication.

Table:13.3.6 Other Control Flags

Serial Interface 0		Serial Interface 1		Serial Interface 2		Detail
Register	Flag	Register	Flag	Register	Flag	
SC0MD2	SC0BRKE	SC1MD2	SC1BRKE	SC2MD2	SC2BRKE	Break status transmission
	SC0BRKF		SC1BRKF		SC2BRKF	Break status reception
	SC0NPE		SC1NPE		SC2NPE	Parity enable
	SC0PM1		SC1PM1		SC2PM1	Additional bit specification
	SC0PM0		SC1PM0		SC2PM0	
	SC0FM1		SC1FM1		SC2FM1	Frame mode specification
	SC0FM0		SC1FM0		SC2FM0	
SC0STR	SC0PEK	SC1STR	SC1PEK	SC2STR	SC2PEK	Parity error detection
	SC0FEF		SC1FEF		SC2FEF	Frame error detection

13.3.2 Clock Synchronous Operation Timing

■ Transmission Timing

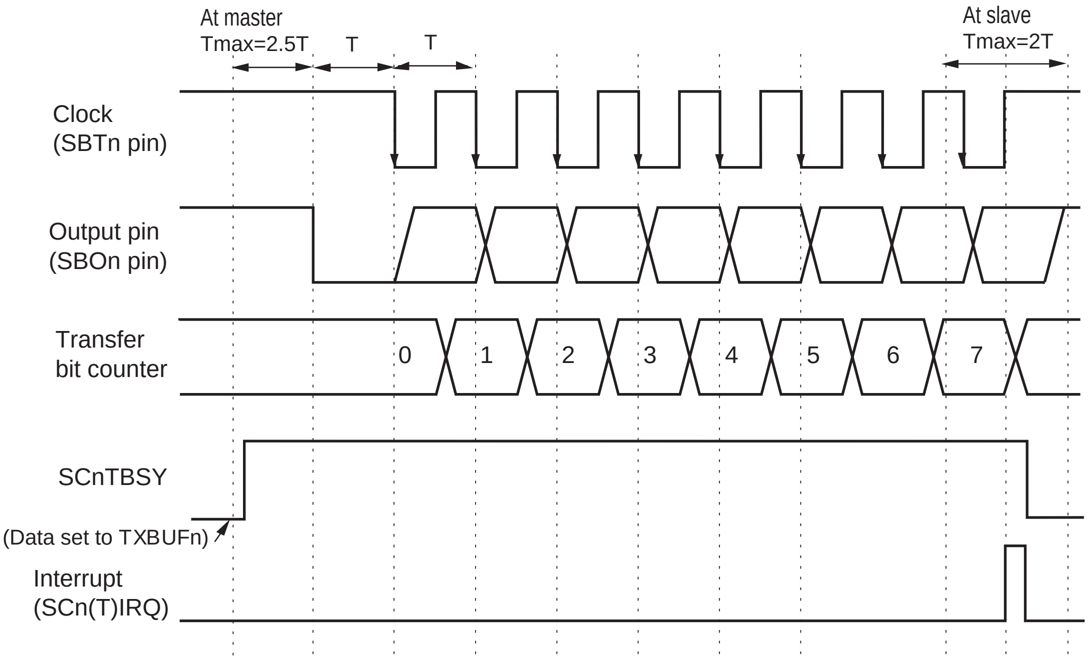


Figure:13.3.6 Transmission Timing (At Falling Edge, Start Condition is Enabled)

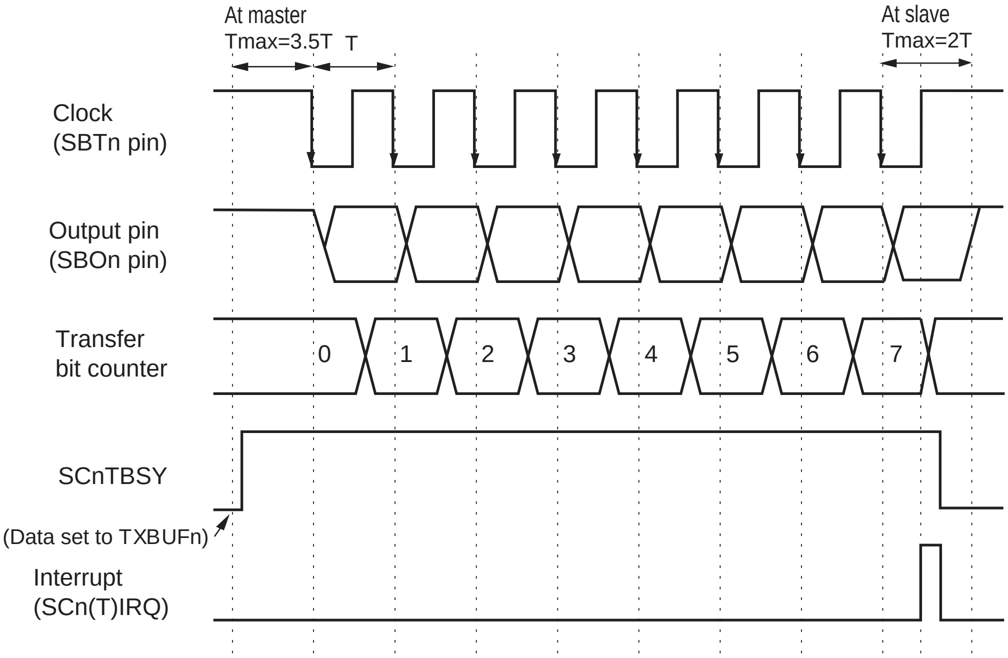


Figure:13.3.7 Transmission Timing (At Falling Edge, Start Condition is Disabled)

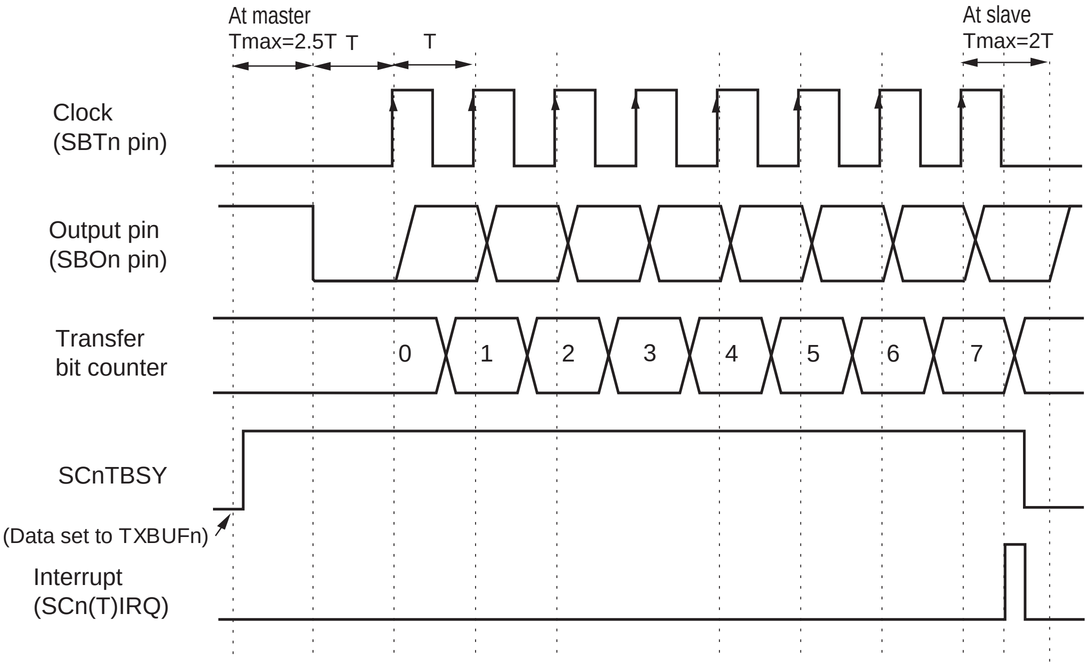


Figure:13.3.8 Transmission Timing (At Rising Edge, Start Condition is Enabled)

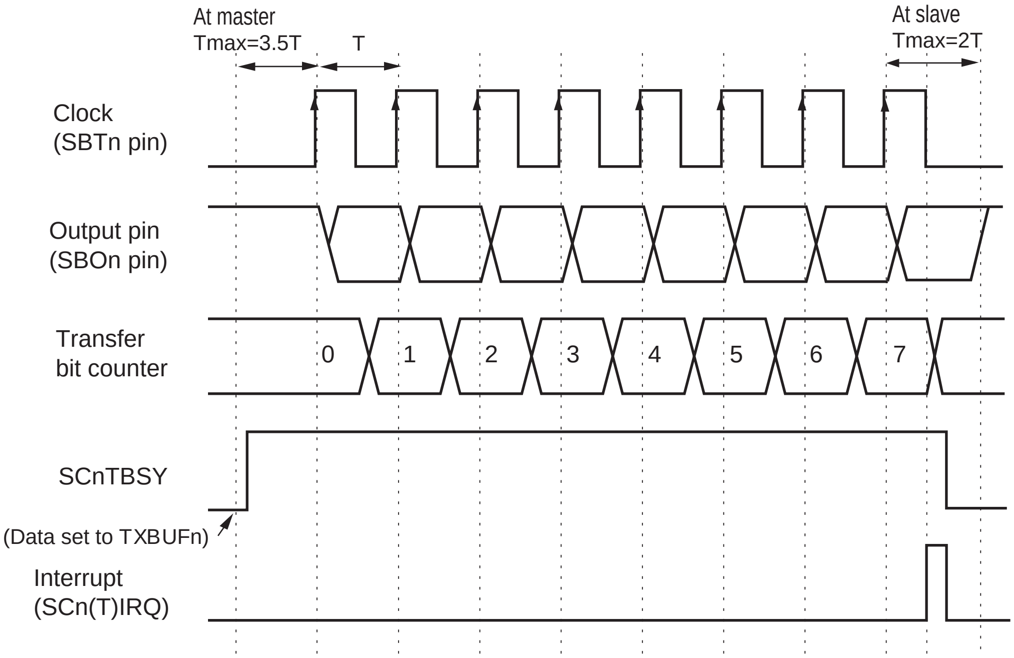


Figure:13.3.9 Transmission Timing (At Rising Edge, Start Condition is Disabled)

■ Reception Timing

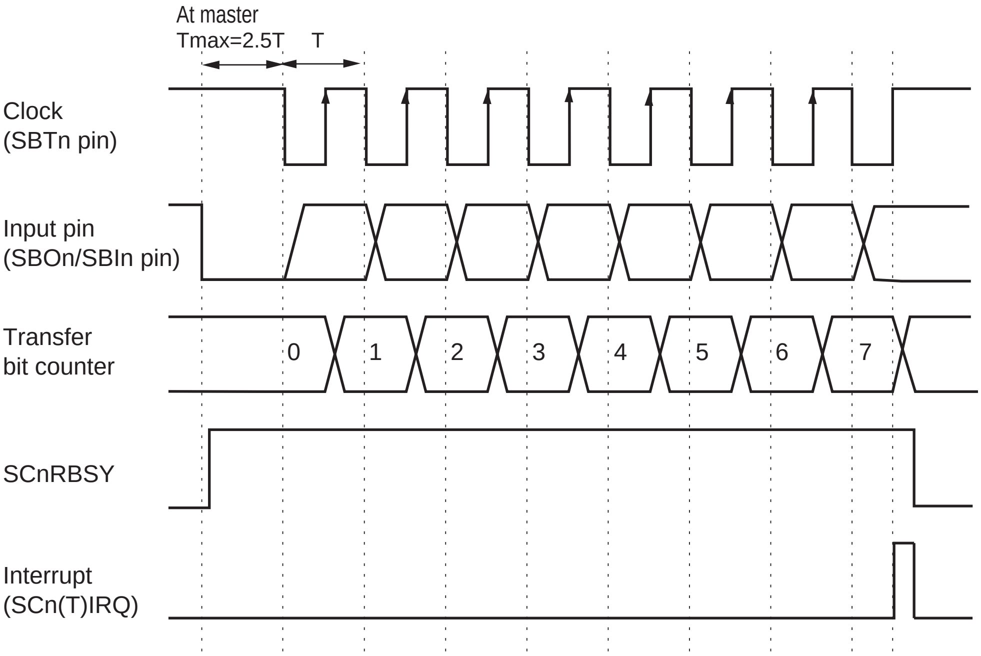


Figure:13.3.10 Reception Timing (At Rising Edge, Start Condition is Enabled)

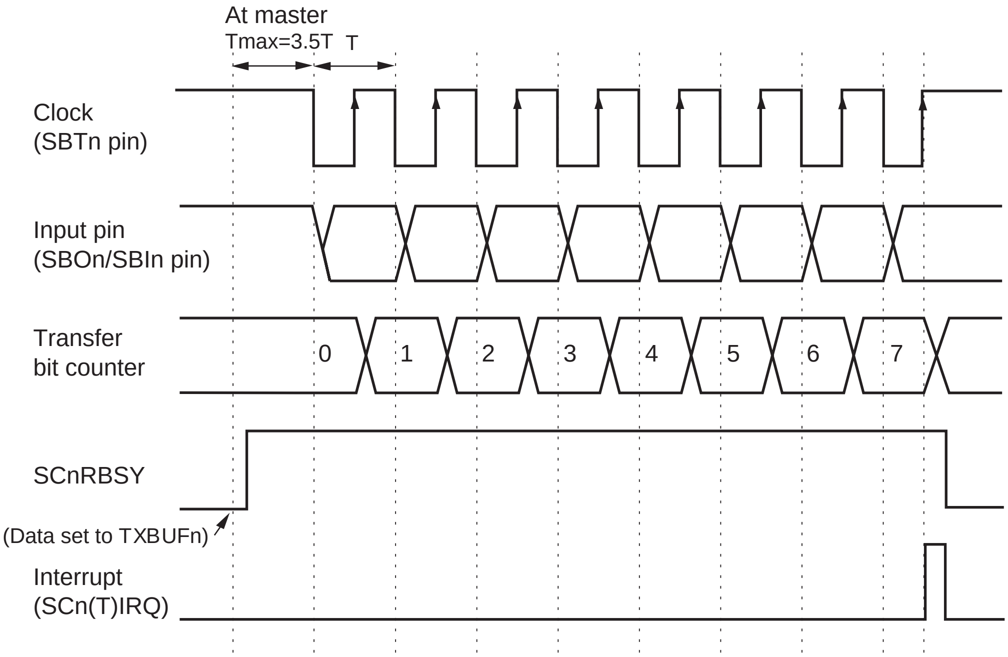


Figure:13.3.11 Reception Timing (At Rising Edge, Start Condition is Disabled)

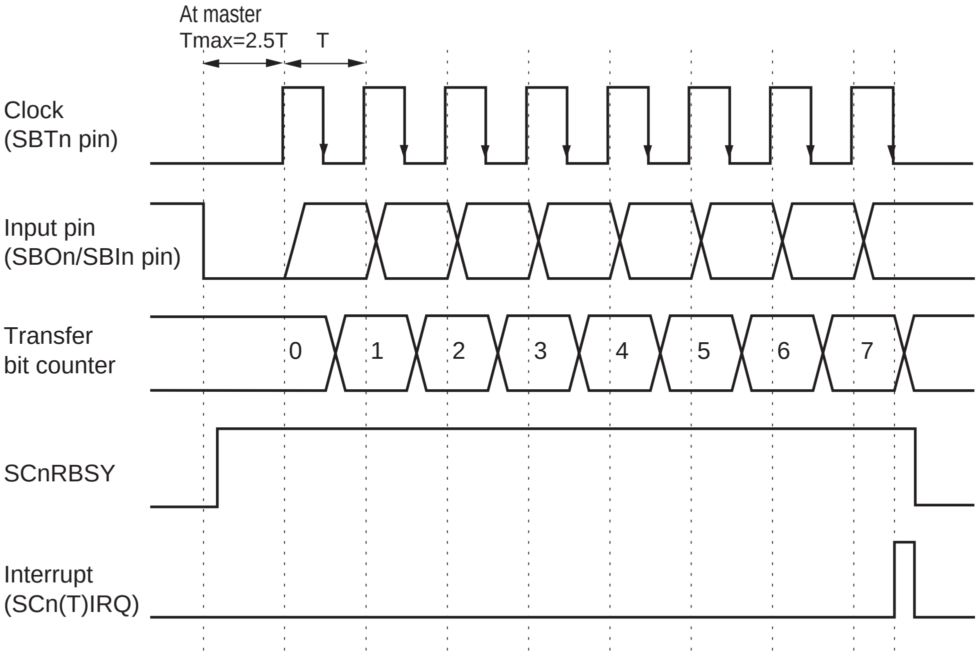


Figure:13.3.12 Reception Timing (At Falling Edge, Start Condition is Enabled)

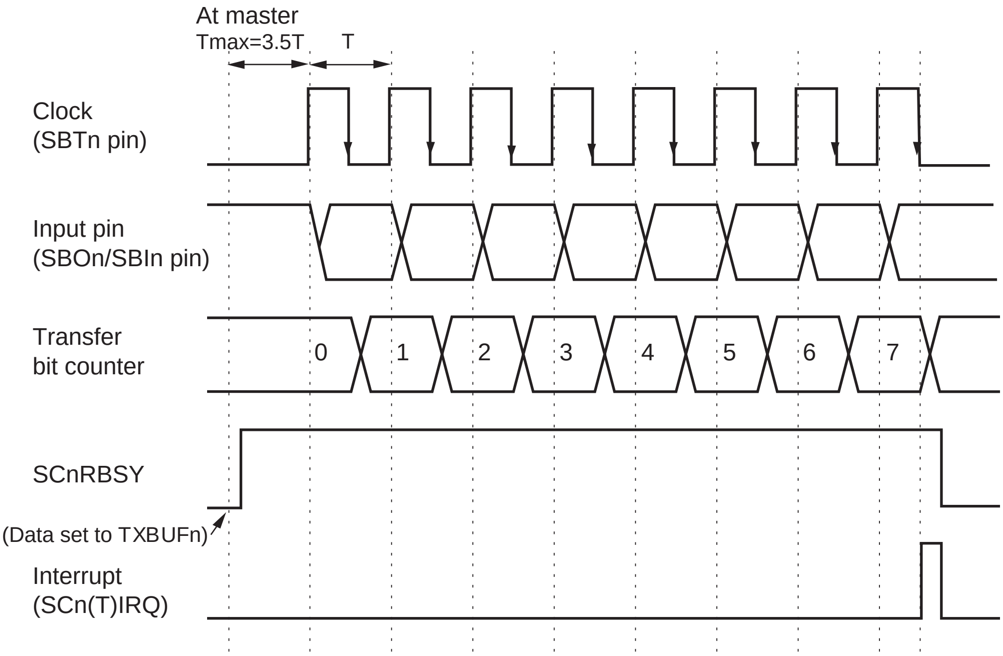


Figure:13.3.13 Reception Timing (At Falling Edge, Start Condition is Disabled)

■ Transmission/Reception Timing 1

When transmission and reception are performed at the same time, the SCnCE1 flag of SCnMD0 register should be set to "0" or "1". Data is received at the opposite edge timing of the output edge of transmission data. Set the polarity of the reception data input edge to be opposite to the output edge of the target transmission data.

Select "start condition disable" when transmission and reception are performed simultaneously.

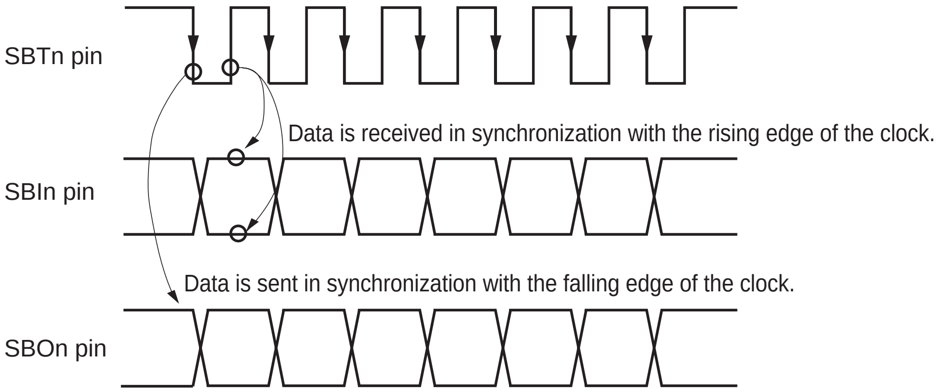

SBTn pin
Data is received in synchronization with the rising edge of the clock.
SBIn pin
Data is sent in synchronization with the falling edge of the clock.
SBOn pin

Figure:13.3.14 Transmission/ Reception Timing
(Reception: at Rising Edge, Transmission: at Falling Edge)

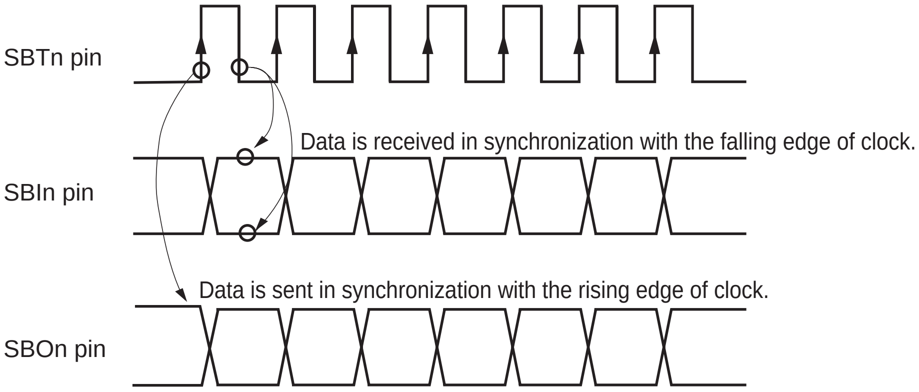

SBTn pin
Data is received in synchronization with the falling edge of clock.
SBIn pin
Data is sent in synchronization with the rising edge of clock.
SBOn pin

Figure:13.3.15 Transmission/Reception Timing
(Reception: at Falling Edge, Transmission: at Rising Edge)

Select "disable start condition" when transmission and reception are performed simultaneously. Otherwise, proper operation cannot be guaranteed.

13.3.3 Communication in STANDBY mode

■ Recovery from STANDBY Mode by Communication Completion Interrupt

This serial interface has the following method for recovering from STANDBY mode.

This serial interface can perform slave reception in STANDBY mode. CPU operation status can be recovered from STANDBY mode to CPU OPERATION mode by Serial Interface n interrupt SCnTIRQ generated after slave reception.

(In STANDBY mode, continuous reception is not available since the next data can not be accepted after the data of the transfer bit count, set by the SCnLNG2 to 0 flags of SCnMD0 register, is received once.) Read the received data from the reception data buffer RXBUFn after recovering to CPU OPERATION mode.

During reception in STANDBY mode, communication with start condition enabled is not available. Disable the start condition. The dummy data should be set in the transmission data buffer TXBUFn before the transition to STANDBY mode.

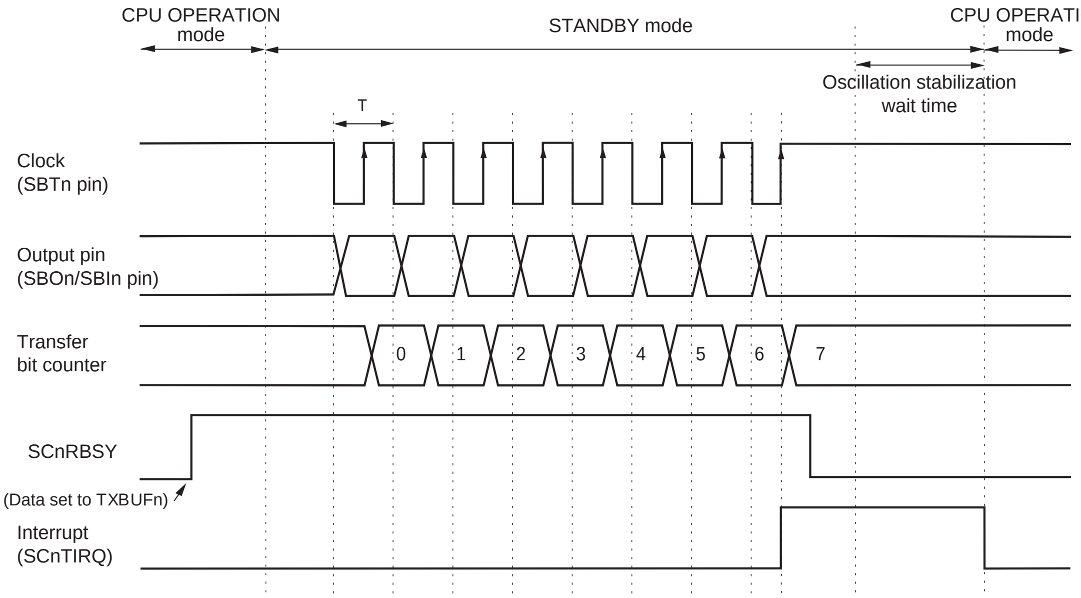


Figure:13.3.16 Reception Timing in STANDBY Mode
(Reception: at Rising Edge, Start Condition is Disabled)

■ Recovery from STANDBY Mode by Reception Interrupt

On Serial Interface 0, 1 and 2, a Serial n UART reception interrupt (SCnRIRQ), Serial Interface 4 a Serial 4 Stop condition interrupt (SC4STPCIRQ) is generated at the falling edge of the serial clock I/O pin (SBTn) by setting the SCnIGC flag of SCINTSEL register to "1".

By this function, the transition to CPU OPERATION mode can be faster than the recovery by a communication completion interrupt when recovering from STANDBY mode to CPU OPERATION mode by serial slave reception.
For example, the differences of operations in recovery from STOP mode are shown below.

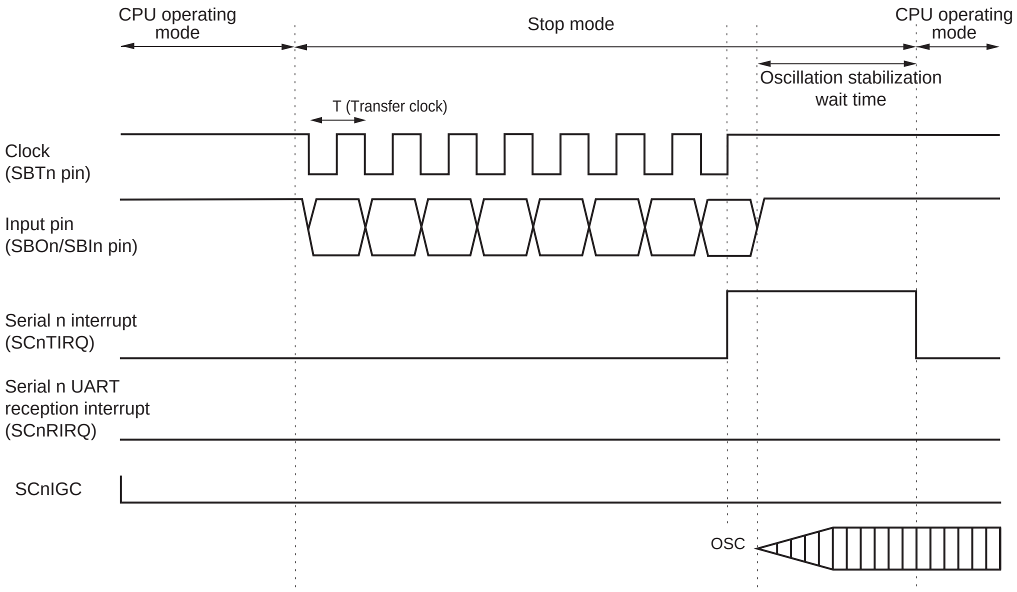


Figure:13.3.17 Recovery Timing from STOP Mode
(SCnIGC=0)

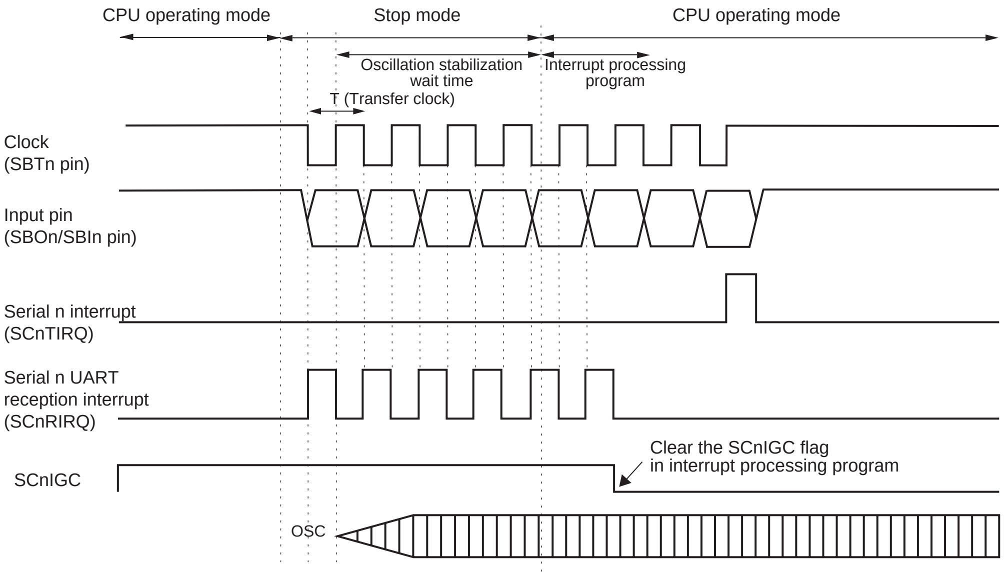


Figure:13.3.18 Recovery Timing from STOP Mode
(SCnIGC=1)

Figure:13.3.4 and Figure:13.3.4 indicate the examples when the serial transfer period is longer than the oscillation stabilization wait time.

To return from STANDBY mode by using Serial Interface n UART reception interrupts / Serial 4 Stop Condition interrupt, execute the following operation in accordance with the procedures below.
1. Clear the SCnICG flag of SCINTSEL register to "0".
2. Clear the MIE flag of PSW register to "0".
3. Set the IRWE flag of MEMCTR register to "1".
4. Clear the SCn(R/S)IR flag of SCn(R/S)ICR register to "0".
(Do not rewrite the SCnRLV1 to 0 flags.)
5. Clear the IRWE flag of MEMCTR register to "0".
Executing the procedures above can avoid multiple executions of interrupt processing program.

SCnIGC flag of clock synchronous serial reception interrupt control register (SCINTSEL) is set to “1” during slave reception in OPERATION mode.
A Serial n UART reception interrupts / Serial 4 Stop Condition interrupt is generated at the falling edge of SBTn before the transition to STANDBY mode from OPERATION mode causing the SCnIGC flag to be cleared.
As a result, The CPU will be stuck in STANDBY mode if the transition to STANDBY mode happened after the Serial Interface n UART reception interrupts / Serial 4 Stop Condition interrupt where the SCnIGC flag is cleared.
To avoid this operation, check program counter (PC) which is saved to the stack during interrupt processing. If this improper operation could occur, rewrite PC of the stack to prevent the transition to STANDBY mode or save the SCnIGC flag and recover it after the interrupt processing.

Recovery from STANDBY mode is available only with synchronous functions. It cannot be performed with UART / IIC function.

13.3.4 Pin Setup

■ Synchronous Serial Interface 0 Pin Setup

Port	Wire system	Type	Pin		Pin setup (flag setup)							
					SC0SEL register	PnDIR register	PnODC register	PnPLUD register	SC0MD1 register			
					Serial 0 I/O pin switching	I/O mode selection	Nch open-drain output selection Arbitrary setting	Pull-up /Pull-down resistor selection Aribitary setting	Serial data input selection	SBT0 pin function selection	Serial input control selection	SBO0 (RXD0) pin function selection
					0:A type 1:B type	0: input mode 1: output mode	0: push/pull 1: Nch open-drain	0: not added 1: added	0: data input from SBI0 1: data input from SBO0	0: port 1: transfer clock I/O	0:"1" input 1: serial input	0: port 1: serial data output
					0SL0	PnDIRm	PnODCm	PnPLUm	SC0IOM	SC0SBTS	SC0SBIS	SC0SBOS
Port 0	3-wire	Transmission only	P03/SBO0A		0	P0DIR3:1	P0ODC3	P0PLU3	0	1	0	1
			-			-	-	-				
			P04/SBT0A	Master		P0DIR4:1	P0ODC4	P0PLU4				
				Slave		P0DIR4:0	-					
		Reception only	-			-	-	-	0	1	1	0
			P02/SBI0A			P0DIR2:0	-	-				
			P04/SBT0A	Master		P0DIR4:1	P0ODC4	P0PLU4				
				Slave		P0DIR4:0	-					
		Transmission/ Reception	P03/SBO0A			P0DIR3:1	P0ODC3	P0PLU3	0	1	1	1
			P02/SBI0A			P0DIR2:0	-	-				
			P04/SBT0A	Master		P0DIR4:1	P0ODC4	P0PLU4				
				Slave		P0DIR4:0	-					
	2-wire	Transmission	P03/SBO0A			P0DIR3:1	P0ODC3	P0PLU3	1	1	0	1
			-			-	-	-				
			P04/SBT0A	Master		P0DIR4:1	P0ODC4	P0PLU4				
				Slave		P0DIR4:0	-					
		Reception	P03/SBO0A			P0DIR3:0	-	-	1	1	1	0
			-			-	-	-				
			P04/SBT0A	Master		P0DIR4:1	P0ODC4	P0PLU4				
				Slave		P0DIR4:0	-					

Port	Wire system	Type	Pin		Pin setup (flag setup)							
					SC0SEL register	PnDIR register	PnODC register	PnPLUD register	SC0MD1 register			
					Serial 0 I/O pin switching	I/O mode selection	Nch open-drain output selection Arbitrary setting	Pull-up /Pull-down resistor selection Aribitary setting	Serial data input selection	SBT0 pin function selection	Serial input control selection	SBO0 (RXD0) pin function selection
					0:A type 1:B type	0: input mode 1: output mode	0: push/pull 1: Nch open-drain	0: not added 1: added	0: data input from SBI0 1: data input from SBO0	0: port 1: transfer clock I/O	0:"1" input 1: serial input	0: port 1: serial data output
					0SL0	PnDIRm	PnODCm	PnPLUm	SC0IOM	SC0SBTS	SC0SBIS	SC0SBOS
Port 4	3-wire	Transmission only	P43/SBO0B		1	P4DIR3:1	P4ODC3	P4PLU3	0	1	0	1
			-			-	-	-				
			P45/SBT0B	Master		P4DIR5:1	P4ODC5	P4PLU5				
				Slave		P4DIR5:0	-					
		Reception only	-			-	-	-	0	1	1	0
			P44/SBI0B			P4DIR4:0	-	-				
			P45/SBT0B	Master		P4DIR5:1	P4ODC5	P4PLU5				
				Slave		P4DIR5:0	-					
		Transmission/ Reception	P43/SBO0B			P4DIR3:1	P4ODC3	P4PLU3	0	1	1	1
			P44/SBI0B			P4DIR4:0	-	-				
			P45/SBT0B	Master		P4DIR5:1	P4ODC5	P4PLU5				
				Slave		P4DIR5:0	-					
	2-wire	Transmission	P43/SBO0B			P4DIR3:1	P4ODC3	P4PLU3	1	1	0	1
			-			-	-	-				
			P45/SBT0B	Master		P4DIR5:1	P4ODC5	P4PLU5				
				Slave		P4DIR5:0	-					
		Reception	P43/SBO0B			P4DIR3:0	-	-	1	1	1	0
			-			-	-	-				
			P45/SBT0B	Master		P4DIR5:1	P4ODC5	P4PLU5				
				Slave		P4DIR5:0	-					

■ Synchronous Serial Interface 1 Pin Setup

Port	Wire system	Type	Pin		Pin setup (flag setup)							
					SC1SEL register	PnDIR register	PnODC register	PnPLUD register	SC1MD1 register			
					Serial 1 I/O pin switching	I/O mode selection	Nch open-drain output selection Arbitrary setting	Pull-up /Pull-down resistor selection Aribitrary setting	Serial data input selection	SBT1 pin function selection	Serial input control selection	SBO1 (RXD1) pin function selection
					0:A type 1:B type	0: input mode 1: output mode	0: push/ pull 1: Nch open-drain	0: not added 1: added	0: data input from SBI1 1: data input from SBO1	0: port 1: transfer clock I/O	0:"1" input 1: serial input	0: port 1: serial data output
					0SL1	PnDIRm	PnODCm	PnPLUm	SC1IOM	SC1SBTS	SC1SBIS	SC1SBOS
Port 5	3-wire	Transmission only	P50/SBO1A		0	P5DIR0:1	P5ODC0	P5PLU0	0	1	0	1
			-			-	-	-				
			P52/ SBT1A	Master		P5DIR2:1	P5ODC2	P5PLU2				
				Slave		P5DIR2:0	-					
		Reception only	-			-	-	-	0	1	1	0
			P51/SBI1A			P5DIR1:0	-	-				
			P52/ SBT1A	Master		P5DIR2:1	P5ODC2	P5PLU2				
				Slave		P5DIR2:0	-					
		Transmission/ Reception	P50/SBO1A			P5DIR0:1	P5ODC0	P5PLU0	0	1	1	1
			P51/SBI1A			P5DIR1:0	-	-				
			P52/ SBT1A	Master		P5DIR2:1	P5ODC2	P5PLU2				
				Slave		P5DIR2:0	-					
	2-wire	Transmission	P50/SBO1A			P5DIR0:1	P5ODC0	P5PLU0	1	1	0	1
			-			-	-	-				
			P52/ SBT1A	Master		P5DIR2:1	P5ODC2	P5PLU2				
				Slave		P5DIR2:0	-					
		Reception	P50/SBO1A			P5DIR0:0	-	-	1	1	1	0
			-			-	-	-				
			P52/ SBT1A	Master		P5DIR2:1	P5ODC2	P5PLU2				
				Slave		P5DIR2:0	-					

Port	Wire system	Type	Pin		Pin setup (flag setup)							
					SC1SEL register	PnDIR register	PnODC register	PnPLUD register	SC1MD1 register			
					Serial 1 I/O pin switching	I/O mode selection	Nch open-drain output selection Arbitrary setting	Pull-up /Pull-down resistor selection Aribitary setting	Serial data input selection	SBT1 pin function selection	Serial input control selection	SBO1 (RXD1) pin function selection
					0:A type 1:B type	0: input mode 1: output mode	0: push/pull 1: Nch open-drain	0: not added 1: added	0: data input from SBI1 1: data input from SBO1	0: port 1: transfer clock I/O	0:”1" input 1: serial input	0: port 1: serial data output
					0SL1	PnDIRm	PnODCm	PnPLUm	SC1IOM	SC1SBTS	SC1SBIS	SC1SBOS
Port 7	3-wire	Transmission only	P75/SBO1B		1	P7DIR5:1	P7ODC5	P7PLU5	0	1	0	1
			-			-	-	-				
			P77/ SBT1B	Master		P7DIR7:1	P7ODC7	P7PLU7				
				Slave		P7DIR7:0	-					
		Reception only	-			-	-	-	0	1	1	0
			P76/SBI1B			P7DIR6:0	-	-				
			P77/ SBT1B	Master		P7DIR7:1	P7ODC7	P7PLU7				
				Slave		P7DIR7:0	-					
		Transmission/ Reception	P75/SBO1B			P7DIR5:1	P7ODC5	P7PLU5	0	1	1	1
			P76/SBI1B			P7DIR6:0	-	-				
			P77/ SBT1B	Master		P7DIR7:1	P7ODC7	P7PLU7				
				Slave		P7DIR7:0	-					
	2-wire	Transmission	P75/SBO1B			P7DIR5:1	P7ODC5	P7PLU5	1	1	0	1
			-			-	-	-				
			P77/ SBT1B	Master		P7DIR7:1	P7ODC7	P7PLU7				
				Slave		P7DIR7:0	-					
		Reception	P75/SBO1B			P7DIR5:0	-	-	1	1	1	0
			-			-	-	-				
			P77/ SBT1B	Master		P7DIR7:1	P7ODC7	P7PLU7				
				Slave		P7DIR7:0	-					

■ Synchronous Serial Interface 2 Pin Setup

Port	Wire system	Type	Pin		Pin setup (flag setup)						
					PnDIR register	PnODC register	PnPLUD register	SC2MD1 register			
					I/O mode selection	Nch open-drain output selection Arbitrary setting	Pull-up /Pull-down resistor selection Aribitary setting	Serial data input selection	SBT1 pin function selection	Serial input control selection	SBO2 (RXD2) pin function selection
					0: input mode 1: output mode	0: push/pull 1: Nch open-drain	0: not added 1: added	0: data input from SBI2 1: data input from SBO2	0: port 1: transfer clock I/O	0:"1" input 1: serial input	0: port 1: serial data output
					PnDIRm	PnODCm	PnPLUm	SC2IOM	SC2SBTS	SC2SBIS	SC2SBOS
Port 6	3-wire	Transmission only	P65/SBO2		P6DIR5:1	P6ODC5	P6PLU5	0	1	0	1
			-		-	-	-				
			P67/SBT2	Master	P6DIR7:1	P6ODC7	P6PLU7				
				Slave	P6DIR7:0	-					
		Reception only	-		-	-	-	0	1	1	0
			P66/SBI2		P6DIR6:0	-	-				
			P67/SBT2	Master	P6DIR7:1	P6ODC7	P6PLU7				
				Slave	P6DIR7:0	-					
		Transmission/ Reception	P65/SBO2		P6DIR5:1	P6ODC5	P6PLU5	0	1	1	1
			P66/SBI2		P6DIR6:0	-	-				
			P67/SBT2	Master	P6DIR7:1	P6ODC7	P6PLU7				
				Slave	P6DIR7:0	-					
	2-wire	Transmission	P65/SBO2		P6DIR5:1	P6ODC5	P6PLU5	1	1	0	1
			-		-	-	-				
			P67/SBT2	Master	P6DIR7:1	P6ODC7	P6PLU7				
				Slave	P6DIR7:0	-					
		Reception	P65/SBO2		P6DIR5:0	-	-	1	1	1	0
			-		-	-	-				
			P67/SBT2	Master	P6DIR7:1	P6ODC7	P6PLU7				
				Slave	P6DIR7:0	-					

■ Synchronous Serial Interface 4 Pin Setup

Port	Wire system	Type	Pin		Pin setup (flag setup)							
					SC0SEL register	PnDIR register	PnODC register	PnPLUD register	SC4MD1 register			
					Serial 0 I/O pin switching	I/O mode selection	Nch open-drain output selection Arbitrary setting	Pull-up /Pull-down resistor selection Aribitary setting	Serial data input selection	SBT0 pin function selection	Serial input control selection	SBO4 (SDA4) pin function selection
					0:A type 1:B type	0: input mode 1: output mode	0: push/ pull 1: Nch open-drain	0: not added 1: added	0: data input from SBI4 1: data input from SBO4	0: port 1: transfer clock I/O	0:"1" input 1: serial input	0: port 1: serial data output
					0SL0	PnDIRm	PnODCm	PnPLUm	SC0IOM	SC0SBTS	SC0SBIS	SC0SBOS
Port 7	3-wire	Trans-mission only	P71/SBO4A		0	P7DIR1:1	P7ODC1	P7PLU1	0	1	0	1
			-			-	-	-				
			P72/ SBT4A	Master		P7DIR2:1	P7ODC2	P7PLU2				
				Slave		P7DIR2:0	-					
		Recep-tion only	-			-	-	-	0	1	1	0
			P70/SBI4A			P7DIR0:0	-	-				
			P72/ SBT4A	Master		P7DIR2:1	P7ODC2	P7PLU2				
				Slave		P7DIR2:0	-					
		Trans-mission/ Recep-tion	P71/SBO4A			P7DIR1:1	P7ODC1	P7PLU1	0	1	1	1
			P70/SBI4A			P7DIR0:0	-	-				
			P72/ SBT4A	Master		P7DIR2:1	P7ODC2	P7PLU2				
				Slave		P7DIR2:0	-					
	2-wire	Trans-mission	P71/SBO4A			P7DIR1:1	P7ODC1	P7PLU1	1	1	0	1
			-			-	-	-				
			P72/ SBT4A	Master		P7DIR2:1	P7ODC2	P7PLU2				
				Slave		P7DIR2:0	-					
		Recep-tion	P71/SBO4A			P7DIR1:0	-	-	1	1	1	0
			-			-	-	-				
			P72/ SBT4A	Master		P7DIR2:1	P7ODC2	P7PLU2				
				Slave		P7DIR2:0	-					

Port	Wire system	Type	Pin		Pin setup (flag setup)							
					SC0SEL register	PnDIR register	PnODC register	PnPLUD register	SC4MD1 register			
					Serial 0 I/O pin switching	I/O mode selection	Nch open-drain output selection Arbitrary setting	Pull-up /Pull-down resistor selection Aribitary setting	Serial data input selection	SBT0 pin function selection	Serial input control selection	SBO4 (SDA4) pin function selection
					0:A type 1:B type	0: input mode 1: output mode	0: push/pull 1: Nch open-drain	0: not added 1: added	0: data input from SBI4 1: data input from SBO4	0: port 1: transfer clock I/O	0:"1" input 1: serial input	0: port 1: serial data output
					0SL0	PnDIRm	PnODCm	PnPLUm	SC0IOM	SC0SBTS	SC0SBIS	SC0SBOS
Port 3	3-wire	Transmission only	P33/SBO4B		1	P3DIR3:1	P3ODC3	P3PLUD3	0	1	0	1
			-			-	-	-				
			P34/ SBT4B	Master		P3DIR4:1	P3ODC4	P3PLUD4				
				Slave		P3DIR4:0	-					
		Reception only	-			-	-	-	0	1	1	0
			P35/SBI4B			P3DIR5:0	-	-				
			P34/ SBT4B	Master		P3DIR4:1	P3ODC4	P3PLUD4				
				Slave		P3DIR4:0	-					
		Transmission/ Reception	P33/SBO4B			P3DIR3:1	P3ODC3	P3PLUD3	0	1	1	1
			P35/SBI4B			P3DIR5:0	-	-				
			P34/ SBT4B	Master		P3DIR4:1	P3ODC4	P3PLUD4				
				Slave		P3DIR4:0	-					
	2-wire	Transmission	P33/SBO4B			P3DIR3:1	P3ODC3	P3PLUD3	1	1	0	1
			-			-	-	-				
			P34/ SBT4B	Master		P3DIR4:1	P3ODC4	P3PLUD4				
				Slave		P3DIR4:0	-					
		Reception	P33/SBO4B			P3DIR3:0	-	-	1	1	1	0
			-			-	-	-				
			P34/ SBT4B	Master		P3DIR4:1	P3ODC4	P3PLUD4				
				Slave		P3DIR4:0	-					

13.3.5 Setup Example

■ Transmission/Reception Setup Example

The setup example for clock synchronous serial communication using Serial Interface 0 is shown. Table:13.3.7 shows the conditions in transmission/reception. The basic setup procedures are the same in Serial Interface 0 to 4. However, pin settings (4) and (5) differ in each serial interface.

Table:13.3.7 Setup Examples for Synchronous Serial Interface Transmission/Reception

Setup item	Set to
Serial data input selection	SBI0
Transfer bit count	8 bits
Start condition	None
First transfer bit	MSB
Input edge	Falling edge
Output edge	Rising edge
Clock	Clock master
Clock source	fs/2
Clock source divide-by-8 (Serial Interface 0, 1 and 2)	Not divided by 8
SBT0/SBO0 pin style	Nch open-drain
SBT0 pin pull-up resistor	Added
SBO0 pin pull-up resistor	Added
Serial 0 interrupt	Enabled
SBO0 output after last data output	Fixed to "High"

An example setup procedure, with a description of each step is shown below

Setup Procedure	Description
(1) Select the prescaler operation SC0MD3(0x03F14) bp3: SC0PSCE =1	(1) Set the SC0PSCE flag of SC0MD3 register to "1" to select "Enable count" for prescaler count control.
(2) Select the clock source SC0MD3(0x03F14) bp2 to 0: SC0PSC2 to 0 =100	(2) Set the SC0PSC2 to 0 flags of SC0MD3 register to "100" to select fs/2 to the clock source.
(3) Control the SBO0 output after the last data output SC0MD3(0x03F14) bp7 to 6: SC0FDC1 to 0 =00	(3) Set the SC0FDC1 to 0 flags of SC0MD3 register to "0, 0" to select output fixed to "High" after the SBO0 last data output.
(4) Control the pin style [set the pin corresponding to each serial] P0ODC(0x03EF0) bp3: P0ODC3 =1 bp4: P0ODC4 =1 P0PLU(0x03EA0) bp3: P0PLU3 =1 bp4: P0PLU4 =1	(4) Set the P0ODC3 and 4 flags of P0ODC register to "11" to select Nch open-drain to SBO0/SBT0. Set the P0PLU3 and 4 flags of P0PLU register to "11" to enable the pull-up resistor. (Set the pin corresponding to serial interface)

Setup Procedure	Description
(5) Control the pin direction [set the pin corresponding to each serial] P0DIR(0x03F90) bp3: P0DIR3 =1 bp4: P0DIR4 =1	(5) Set the P0DIR3 and 4 flags of P0DIR register to "11" to set P03 and P04 to output mode and P02 to input mode. (Set the pin corresponding to serial interface)
(6) Set SC0MD0 register Select the transfer bit count SC0MD0(0x03F11) bp2 to 0: SC0LNG2 to 0 =111	(6) Set the SC0LNG2 to 0 flags of Serial Interface 0 mode register 0 (SC0MD0) to "111" to set the transfer bit count to 8 bits.
Select the start condition SC0MD0(0x03F11) bp3: SC0STE =0	Set the SC0STE flag of SC0MD0 register to "0" to disable the start condition.
Select the transfer first bit SC0MD0(0x03F11) bp4: SC0DIR =0	Set the SC0DIR flag of SC0MD0 register to "0" to set MSB as the transfer first bit.
Select the transfer edge SC0MD0(0x03F11) bp7: SC0CE1 =1	Set the SC0CE1 flag of SC0MD0 register to "1" to set the transmission data input edge to "rising" and the reception data output edge to "falling".
(7) Set SC0MD1 register Select the communication type SC0MD1(0x03F12) bp0: SC0CMD =0	(7) Set the SC0CMD flag of SC0MD1 register to "0" to select the clock synchronous.
Select the transfer clock SC0MD1(0x03F12) bp2: SC0MST =1 bp3: SC0CKM =0 (Serial Interface 0, 1 and 2)	Set the SC0MST flag of SC0MD1 register to "1" to select the clock master (internal clock); and, set the SC0CKM flag to "0" to select "the source clock not divided by 8" for Serial Interface 0, 1 and 2.
Control the pin function SC0MD1(0x03F12) bp4: SC0SBOS =1 bp5: SC0SBIS =1 bp6: SC0SBTS =1 bp7: SC0IOM =0	Set the SC0SBOS, SC0SBIS and SC0SBTS flags of SC0MD1 register to "1" to set SBO0 pin to the serial data output, SBI0 pin to the serial data input and SBT0 pin to the serial clock input/output. Set the SC0IOM flag to "0" to set the serial data input from SBI0 pin.
(8) Set the interrupt level PSW bp6: MIE =0 SC0TICR(0x03FFB) bp7 to 6: SC0LV1 to 0 =10	(8) Clear the MIE flag of PSW to “0” to disable all maskable interrupts. Set the interrupt level by the SC0LV1 to 0 flags of Serial 0 interrupt control register (SC0TICR).
(9) Enable the interrupt SC0TICR (0x03FFB) bp1: SC0TIE =1 PSW bp6: MIE =1	(9) Set the SC0TIE flag of SC0TICR register to "1" to enable the interrupt. If any interrupt request flag (SC0TIR of SC0TICR register) is already set, clear SC0TIR prior to enabling the interrupt. Set the MIE flag of PSW to “1” to enable maskable interrupts.

Setup Procedure	Description
(10) Start the serial transmission Transmission data→TXBUF0 (0x03F17) Reception data→Input to SBI0 pin	Set the transmission data to the serial transmission data buffer TXBUF0. The transfer clock is generated and transmission or reception is started. When transmission is finished, the Serial Interface 0 interrupt SC0TIRQ is generated. [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup]

*Each setup in (1) to (3), (6) to (7) and (8) to (9) can be set at the same time.

Set the SCnSBIS of SCnMD1 register to "0" and select a port in order to operate only transmission with 3-wire system. Set the SCnSBOS of SCnMD1 register to "0" and select a port in order to operate only reception.

In a 2-wire system communicating using SBOn pin, serial data is input and output via SBOn pin. Port direction control register controls the switching between input and output. SCnSBIS of SCnMD1 register must be set to "1" to select "serial data input". SBIn pin can be used as a general-purpose port.

This serial interface has an emergency reset function. If the communication is needed to be stopped by force, set SCnSBOS and SCnSBIS of SCnMD1 register to "0".

Set each flag in accordance with the procedures indicated in the table above. Be sure to start communication after the settings of all control registers (refer to Table:13.2.1 but exclude TXBUFn, RXBUFn and SCnTRB) are completed.

The transfer rate must be under 5.0 MHz for setting a transfer clock by SCnMD3 register.

■ Reception Setup Example (STANDBY Mode Reception)

The following shows the setup example for STANDBY mode reception of clock synchronous serial communication using Serial Interface 0. Table:13.3.8 shows the condition at reception. The basic procedures are the same in Serial Interface 0 to 4. Pin settings (4) and (5) differ in each serial interface.

Table:13.3.8 Setup Examples for Synchronous Serial Interface Reception

Setup item	Set to
Serial data input selection	SBI0
Transfer bit count	8 bits
Start condition	None
First transfer bit	MSB
Input edge	Falling edge
Output edge	Rising edge
Clock	Clock slave
Clock source	fs/2
Clock source divide-by-8 (Serial Interface 0, 1 and 2)	Not divided by 8
SBT0/SBO0 pin style	Nch open-drain
SBT0 pin pull-up resistor	Added
SBO0 pin pull-up resistor	Added
Serial 0 interrupt	Enabled
SBO0 output after last data output	Fixed to "High"

An example setup procedure, with a description of each step is shown below

Setup Procedure	Description
(1) Select the prescaler operation SC0MD3 (0x03F14) bp3: SC0PSCE =1	(1) Set the SC0PSCE flag of SC0MD3 register to "1" to select "Enable count" for prescaler count control.
(2) Select the clock source SC0MD3 (0x03F14) bp2 to 0: SC0PSC2 to 0 =100	(2) Set the SC0PSC2 to 0 flags of SC0MD3 register to "100" to select fs/2 to the clock source.
(3) Control the SBO0 output after the last data output SC0MD3 (0x03F14) bp7 to 6: SC0FDC1 to 0 =00	(3) Set the SC0FDC1 to 0 flags of SC0MD3 register to "0, 0" to select output fixed at "High" after the SBO0 last data output.
(4) Control the pin style [set the pin corresponding to each serial] P0ODC(0x03EF0) bp4: P0ODC4 =1 P0PLU(0x03EA0) bp4: P0PLU4 =1	(4) Set the P0ODC4 flags of P0ODC register to "1" to select Nch open-drain to SBT0. Set the P0PLU4 flag of P0PLU register to "1" to enable the pull-up resistor. (Set the pin corresponding to each serial interface)
(5) Control the pin direction [set the pin corresponding to each serial] P0DIR(0x03F93) bp2: P0DIR6 =0 bp4: P0DIR5 =0	(5) Set the P0DIR2 flag of P0DIR register to “0” and set the P0DIR4 flag to "0" to set P02 and P04 to input mode (Set the pin corresponding to each serial interface)

Setup Procedure	Description
(6) Set SC0MD0 register Select the transfer bit count SC0MD0 (0x03F11) bp2 to 0: SC0LNG2 to 0 =111	(6) Set the SC0LNG2 to 0 flags of Serial Interface 0 mode register 0 (SC0MD0) to "111" to set the transfer bit count to 8 bits.
Select the start condition SC0MD0 (0x03F11) bp3: SC0STE =0	Set the SC0STE flag of SC0MD0 register to "0" to disable the start condition.
Select the transfer first bit SC0MD0 (0x03F11) bp4: SC0DIR =0	Set the SC0DIR flag of SC0MD0 register to "0" to select MSB as the transfer first bit.
Select the transfer edge SC0MD0 (0x03F11) bp7: SC0CE1 =1	Set the SC0CE1 flag of SC0MD0 register to "1" to set the transmission data input edge to "rising" and the reception data output edge to "falling".
(7) Set SC0MD1 register Select the communication type SC0MD1 (0x03F12) bp0: SC0CMD =0	(7) Set the SC0CMD flag of SC0MD1 register to "0" to select the clock synchronous.
Select the transfer clock SC0MD1 (0x03F12) bp2: SC0MST =0 bp3: SC0CKM =0 (Serial Interface 0, 1 and 2)	Set the SC0MST flag of SC0MD1 register to "0" to select the clock slave (external clock). Set the SC0CKM flag to "0" to select "the source clock not divided by 8" for Serial Interface 0, 1 and 2.
Control the pin function SC0MD1 (0x03F12) bp4: SC0SBOS =0 bp5: SC0SBIS =1 bp6: SC0SBTS =1 bp7: SC0IOM =0	Set the SC0SBIS and SC0SBTS flags of SC0MD1 register to "1" to set SBI0 pin to the serial input and SBT0 pin to the transfer clock input/output. Set the SC0IOM flag "0" to set the serial data input from SBI0 pin. Set the SC0SBOS flag to “0” to select the port as SBO0 pin function.
(8) Set the interrupt level PSW bp6: MIE =0 SC0TICR (0x03FFB) bp7 to 6: SC0LV1 to 0 =10	(8) Clear the MIE flag of PSW to “0” to disable all maskable interrupts. Set the interrupt level by the SC0LV1 to 0 flags of Serial 0 interrupt control register (SC0TICR).
(9) Enable the interrupt SC0TICR (0x03FFB) bp1: SC0TIE =1 PSW bp6: MIE =1	(9) Set the SC0TIE flag of SC0TICR register to "1" to enable the interrupt. If any interrupt request flag (SC0TIR of SC0TICR register) is already set, clear SC0TIR prior to enabling the interrupt. Set the MIE flag of PSW to “1” to enable maskable interrupts.
(10) Start the serial transmission Dummy data → TXBUF0 (0x03F17)	(10) Set dummy data to the serial transmission data buffer TXBUF0.
(11) Transfer to STOP mode CPUM(0x03F00) bp3: STOP =1	(11) Set the STOP flag of CPUM register to "1" to transfer to STOP mode.
(12) Start the serial communication Transmission clock → input SBT0 pin Received data → input SBI0 pin	(12) Input the transfer clock to SBT0 pin and transfer data to SBI0 pin.

Setup Procedure	Description
(13) Return from STANDBY mode	(13) The Serial Interface 1 interrupt SC0TIRQ is generated at the same time of the 8th bits data reception. CPU is then returned from STOP mode to NORMAL mode after the oscillation stabilization wait.

*Each setup (1) to (3) and (6) to (8) can be set at the same time.

Disable the start condition during slave reception in STANDBY mode. If the start condition is enabled, proper reception may not be guaranteed.

In a 2-wire system communicating using SBOn pin, serial data is input and output via SBOn pin. Port direction control register controls the switching between input and output. SCnSBIS of SCnMD1 register must be set to "1" to select "serial data input". SBIn pin can be used as a general-purpose port.

This serial interface has an emergency reset function. If the communication is needed to be stopped by force, set SCnSBOS and SCnSBIS of SCnMD1 register to "0".

Set each flag in accordance with the procedures indicated in the table above. Be sure to start communication after the settings of all control registers (refer to Table:13.2.1 but exclude TXBUFn, RXBUFn and SCnTRB) are completed.

The transfer rate must be under 5.0 MHz for setting a transfer clock by SCnMD3 register.

Insert three NOP instructions immediately after the transition to HALT or STOP mode.

13.4 Duplex UART Serial Interface

13.4.1 Operation

Serial Interface 0, 1 and 2 can be used for duplex UART communication.

When setting this serial interface communication mode to “UART”, set Serial Interface n mode register 1 (SCnMD1) while the serial data input pin is at “High”.

"n" = 0, 1, 2 for Serial Interface 0, 1 and 2 respectively in [Chapter 13 13.4.1 Operation].

■ Communication Type

The communication mode can be selected from 2-wire type or 1-wire type. Set the communication mode by the SCnIOM flag of the SCnMD1 register. In 1-wire type, a single pin of the data I/O pin (SBOn) is used, and the data input pin (SBIn) can be used as a general-purpose port. In 1-wire reception, select "serial data input" by setting the SCnSBIS flag of the SCnMD1 register to "1".

■ Activation Factor for Communication

Setting data in the transmission data buffer TXBUFn generates a start condition, and transmission will start. Receiving a start condition starts reception. In reception, when the data length of "Low" for the start bit is longer than 0.5 bit, a start condition is recognized.

■ Transmission

When data is set in the transmission data buffer TXBUFn, transmission is automatically started. When the transmission is completed, the serial n transmission complete interrupt SCnTIRQ is generated.

■ Reception

When a start condition is recognized, reception is started after the transfer bit counter that counts transfer bits is cleared. When reception is completed, the serial n reception complete interrupt SCnRIRQ is generated.

■ Duplex communication

On duplex communication, transmission and reception can be executed independently at the same time. The frame mode and parity bit of the data used on transmission/reception should have the same polarity.

■ Transfer Bit Count Setup

The transfer bit count is automatically set when the frame mode is specified by the SCnFM1 to 0 flags of SCnMD2 register. If the SCnCMD flag of SCnMD1 register is set to "1" and UART communication is selected, the setting of the synchronous serial transfer bit count selection flags SCnLNG2 to 0 of SCnMD0 register is no longer valid.

■ Data Input Pin Setup

Communication mode can be selected from 2-wire [data output pin (TXDn pin), data input pin (RXDn pin)] mode and 1-wire (data I/O pin TXDn) mode. Set communication mode by the SCnIOM flag of SCnMD1 register.
RXDn pin can only be used for serial data input. TXDn pin can be used for serial data input or output.
If "data input from TXDn pin" is selected, the communication mode is 1-wire. Transmission and reception can be switched by the direction control of TXDn pin with I/O selection of the port direction control register.
At this time, RXDn pin can be used as a general-purpose port.

■ Reception Buffer Empty Flag

When Serial Interface n reception complete interrupt SCnRIRQ is generated, data is automatically stored from internal shift register to RXBUFn. If data is stored in shift register RXBUFn, the reception buffer empty flag SCnREMP of SCnSTR register is set to "1" indicating that the reception data is waiting to be read.

SCnREMP is cleared to "0" by reading the RXBUFn data

■ Reception BUSY Flag

When a start condition is recognized, the SCnRBSY flag of SCnSTR register is set to "1". It is cleared to "0" after Serial Interface n reception complete interrupt SCnRIRQ is generated. If the SCnSBIS flag is set to "0" during reception, the SCnRBSY flag is cleared to "0".

■ Transmission BUSY Flag

When data is set in TXBUFn, the SCnTBSY flag of SCnSTR register is set to "1". It is cleared to "0" after Serial Interface n transmission complete interrupt SCnTIRQ is generated. During continuous communication, the set SCnTBSY flag is retained. If the transmission buffer empty flag SCnTEMP is "0" when Serial Interface n transmission complete interrupt SCnTIRQ is generated, the SCnTBSY is cleared to "0". If the SCnSBOS flag is set to "0" during transmission, the SCnTBSY flag is cleared to "0".

■ Frame Mode and Parity Check Setup

The data format at UART communication is shown below.

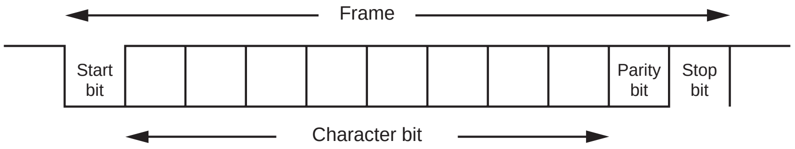


Figure:13.4.1 Transmission/Reception Data Format of UART Serial Interface

The transmission/reception data consists of start bit, character bit, parity bit and stop bit. Table:13.4.1 shows the types of data that can be set.

Table:13.4.1 UART Serial Interface Transmission/Reception Data

Start bit	1 bit
Character bit	7, 8 bits
Parity bit	fixed to “0”, fixed to “1”, odd, even, none
Stop bit	1, 2 bits

The frame mode is set by the SCnFM1 to 0 flags of SCnMD2 register. Table:13.4.2 shows the types of frame mode that can be set. If the SCnCMD flag of SCnMD1 register is set to "1" and UART communication is selected, the transfer bit counts of the SCnLNG2 to 0 flags of SCnMD0 register are no more valid.

Table:13.4.2 UART Serial Interface Frame Mode

SCnMD2 register		Frame mode
SCnFM1	SCnFM0	
0	0	Character bit 7 bits + Stop bit 1 bit
0	1	Character bit 7 bits + Stop bit 2 bits
1	0	Character bit 8 bits + Stop bit 1 bit
1	1	Character bit 8 bits + Stop bit 2 bits

The parity bit is for detecting wrong bits of transmission/reception data. Table:13.4.3 shows the types of the parity bit. The parity bit is set by the SCnNPE and SCnPM1 to 0 flags of SCnMD2 register.

Table:13.4.3 Parity Bit of UART Serial Interface

SCnMD2			Parity bit	Setup
SCnNPE	SCnPM1	SCnPM0		
0	0	0	Fixed at 0	Set parity bit to "0"
0	0	1	Fixed at 1	Set parity bit to "1"
0	1	0	Odd parity	Control the total of "1" of parity bit and character bit to be odd
0	1	1	Even parity	Control the total of "1" of parity bit and character bit to be even
1	-	-	None	Do not add parity bit

- Break Status Transmission Control Setup

The SCnBRKE flag of SCnMD2 register generates the break status. If SCnBRKE is set to "1" to select the break transmission and all bits from start bit to stop bit are transferred "0".

- Reception Error

At reception, there are 3 types of errors: overrun error, parity error and framing error. The reception error can be determined by checking the SCnORE, SCnPEK and SCnFEF flags of SCnSTR register. If one of these flags has an error, the SCnERE flag of SCnSTR register is set to "1". The SCnPEK and SCnFEF flags of the reception error flags are renewed during generation of Serial Interface n reception complete interrupt SCnRIRQ. The SCnORE flag is cleared simultaneously when the next SCnRIRQ is generated after the RXBUFn data is read. The reception error determination should be operated before the next communication is completed. Those error flags have no effects on communication operation. Table:13.4.4 shows reception error factors of reception errors.

Table:13.4.4 Reception Error Factors of UART Serial Interface

Flag	Reception error		
SCnORE	Overrun error	Next data is received before reading the reception buffer	
SCnPEK	Parity error	At fixed to 0	when parity bit is "1"
		At fixed to 1	When parity bit is "0"
		Odd parity	When the total of "1" of parity bit and character bit is even
		Even parity	When the total of "1" of parity bit and character bit is odd
SCnFEF	Framing error	Stop bit is not detected	

■ Determination of Break Status Reception

Reception at break status can be determined by the SCnBRKF flag. If all received data from the start bit and the stop bit are "0", the SCnBRKF flag of SCnMD2 register is set indicating that the break status is generated. The SCnBRKF flag is set when Serial Interface n reception complete interrupt SCnRIRQ is generated.

■ Continuous Communication

This serial interface has a continuous transfer function. If data is set in the transmission data buffer TXBUFn during transmission, the transmission buffer empty flag SCnTEMP is set and continue transmission is operated automatically. There are no communication blanks between current transfer and next transfer. Set the next data to TXBUFn between the time when the data in the TXBUFn is loaded to the transmission shift register and before Serial Interface n transmission complete interrupt SCnTIRQ is generated.

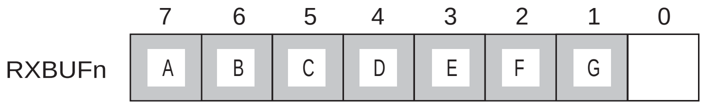
■ Clock Setup

A transfer clock is not necessary at UART communication, but the clock setup is necessary for determining the data transmission/reception timing within the serial interface. Select the timer used as a baud rate timer by the SCnMD3 register.

Be sure to set SCnSBIS and SCnSBOS flags of SCnMD1 to “0” before changing a clock.

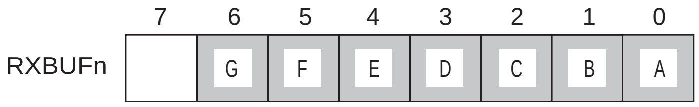
■ Reception Bit Count and First Transfer Bit

In reception, when transfer bit counts = 7 bits, the data storing method for the reception data buffer RXBUFn differs depending on the first transfer bit specification. At MSB first, data are stored in the upper bits of RXBUFn. When transfer bit counts = 7 bits, as shown in Figure:13.4.2, with "G" to be the first bit to transfer and "A" to be the last bit to transfer, data "A" to "G" are stored in bp7 to bp1 of RXBUFn. At LSB first, data are stored in the lower bits of RXBUFn. When transfer bit counts = 7 bits, as shown in Figure:13.4.3, with "G" to be the first bit to transfer and "A" to be the last bit to transfer, data "A" to "G" are stored in bp0 to bp6 of RXBUFn.



	7	6	5	4	3	2	1	0
RXBUFn	A	B	C	D	E	F	G	

Figure:13.4.2 Reception Bit Count and First Transfer Bit (At MSB First)



	7	6	5	4	3	2	1	0
RXBUFn		G	F	E	D	C	B	A

Figure:13.4.3 Reception Bit Count and First Transfer Bit (At LSB First)

■ Transfer Rate Setup (Serial Interface 0, 1 and 2)

The transfer speed can be set using a baud rate timer. The setup example for the transfer speed is shown below.

Table:13.4.5 UART Serial Interface Transfer Speed

	Setup	Register	Page
Serials 0, 1 and 2	Serial Interface 0 and 2 clock source (Timer 0 to 3 or A output)	SC0MD3	XIII-25
		SC1MD3	XIII-25
		SC2MD3	XIII-25
	Timer 0 clock source	TM0MD	VI-12
	Timer 1 clock source	TM1MD	VI-13
	Timer 2 clock source	TM2MD	VI-14
	Timer 3 clock source	TM3MD	VI-15
	Timer A clock source	TMAMD1	VII-6
	Timer 0 compare register	TM0OC	VI-11
	Timer 1 compare register	TM1OC	VI-11
	Timer 2 compare register	TM2OC	VI-11
	Timer 3 compare register	TM3OC	VI-11
	Timer A compare register	TMAOC	VII-5

Timer compare register is set as follows;

overflow cycle = (set value of compare register + 1) × timer clock cycle

baud rate = 1 / (overflow cycle × 2 ×8)("8" is divide-by-8 clock source)

therefore,

set value of compare register = timer clock frequency/(baud rate × 2 × 8) - 1

For example, when setting the baud rate to 300 bps at clock source fs/4 (fpll-div = 8 MHz, fs = fpll-div/2), the set value should be as follows:

set value of compare register = $(8 \times 10^6/2/4)/(300 \times 2 \times 8) - 1$

$= 207$

$= \text{0xCF}$

The following pages show the timer clock source at the standard transfer rate and the set value of compare register when fs=fpll-div/2.

Transfer rate should be selected under 300 kbps.

When the SCnCMD flag of SCnMD1 register is set to “1” to select “Duplex UART”, the transfer clock is divided regardless of the SCnCKM flag.

Table:13.4.6 Setup Value of UART Serial Interface Transfer Speed (decimal) Clock Source Divided by 8

Transfer speed (bps)											
fpll-div (MHz)	Clock source (timer)	300		960		1200		2400		4800	
		Set value	Calculated value	Set value	Calculated value	Set value	Calculated value	Set value	Calculated value	Set value	Calculated value
4.00	fpll-div	-	-	-	-	207	1202	103	2404	51	4808
	fpll-div/4	207	300	64	962	51	1202	25	2404	12	4808
	fpll-div/16	51	300	-	-	12	1202	-	-	-	-
	fpll-div/32	25	300	-	-	-	-	-	-	-	-
	fpll-div/64	12	300	-	-	-	-	-	-	-	-
	fs/2	207	300	64	962	51	1202	25	2404	12	4808
	fs/4	104	297	-	-	25	1202	12	2404	-	-
4.19	fpll-div	-	-	-	-	217	1201	108	2403	54	4761
	fpll-div/4	217	300	67	963	-	-	-	-	-	-
	fpll-div/16	-	-	16	963	-	-	6	2338	-	-
	fpll-div/32	-	-	-	-	-	-	-	-	-	-
	fpll-div/64	-	-	-	-	-	-	-	-	-	-
	fs/2	217	300	67	963	-	-	-	-	-	-
	fs/4	108	300	33	963	-	-	13	2338	-	-
8.00	fpll-div	-	-	-	-	-	-	207	2404	103	4808
	fpll-div/4	-	-	129	962	103	1202	51	2404	25	4808
	fpll-div/16	103	300	-	-	25	1202	12	2404	-	-
	fpll-div/32	51	300	-	-	12	1202	-	-	-	-
	fpll-div/64	25	300	-	-	-	-	-	-	-	-
	fs/2	-	-	129	962	103	1202	51	2404	25	4808
	fs/4	207	300	64	962	51	1202	25	2404	12	4808
8.38	fpll-div	-	-	-	-	-	-	217	2403	108	4805
	fpll-div/4	-	-	135	963	108	1201	-	-	-	-
	fpll-div/16	108	300	33	963	-	-	13	2338	-	-
	fpll-div/32	-	-	16	963	-	-	16	2338	-	-
	fpll-div/64	-	-	-	-	-	-	-	-	-	-
	fs/2	-	-	135	963	108	1201	-	-	-	-
	fs/4	217	300	67	963	-	-	-	-	-	-
12.00	fpll-div	-	-	-	-	-	-	-	-	155	4808
	fpll-div/4	-	-	194	962	155	1202	77	2404	38	4808
	fpll-div/16	155	300	-	-	38	1202	-	-	-	-
	fpll-div/32	77	300	-	-	-	-	-	-	-	-
	fpll-div/64	38	300	-	-	-	-	-	-	-	-
	fs/2	-	-	194	962	155	1202	77	2404	38	4808
	fs/4	-	-	-	-	77	1202	38	2404	-	-
16.00	fpll-div	-	-	-	-	-	-	-	-	207	4808
	fpll-div/4	-	-	-	-	207	1202	103	2404	51	4808
	fpll-div/16	207	300	64	962	51	1202	25	2404	12	4808
	fpll-div/32	103	300	-	-	25	1202	12	2404	-	-
	fpll-div/64	51	300	-	-	12	1202	-	-	-	-
	fs/2	-	-	-	-	207	1202	103	2404	51	4808
	fs/4	-	-	129	962	103	1202	51	2404	25	4808
20.00	fpll-div	-	-	-	-	-	-	-	-	-	-
	fpll-div/4	-	-	-	-	-	-	129	2404	64	4808
	fpll-div/16	-	-	-	-	64	1202	-	-	-	-
	fpll-div/32	129	300	-	-	-	-	-	-	-	-
	fpll-div/64	64	300	-	-	-	-	-	-	-	-
	fs/2	-	-	-	-	-	-	129	2404	64	4808
	fs/4	-	-	162	959	129	1202	64	2404	-	-

Table:13.4.7 Setup Value of UART Serial Interface Transfer Speed (decimal) Clock Source Divided by 8

		Transfer speed (bps)									
fpll-div (MHz)	Clock source (timer)	9600		19200		28800		31250		38400	
		Set value	Calculated value	Set value	Calculated value	Set value	Calculated value	Set value	Calculated value	Set value	Calculated value
4.00	fpll-div	25	9615	12	19231	-	-	7	31250	-	-
	fpll-div/4	-	-	-	-	-	-	1	31250	-	-
	fpll-div/16	-	-	-	-	-	-	-	-	-	-
	fpll-div/32	-	-	-	-	-	-	-	-	-	-
	fpll-div/64	-	-	-	-	-	-	-	-	-	-
	fs/2	-	-	-	-	-	-	1	31250	-	-
	fs/4	-	-	-	-	-	-	-	-	-	-
4.19	fpll-div	26	9699	-	-	-	-	-	-	-	-
	fpll-div/4	-	-	-	-	-	-	-	-	-	-
	fpll-div/16	-	-	-	-	-	-	-	-	-	-
	fpll-div/32	-	-	-	-	-	-	-	-	-	-
	fpll-div/64	-	-	-	-	-	-	-	-	-	-
	fs/2	-	-	-	-	-	-	-	-	-	-
	fs/4	-	-	-	-	-	-	-	-	-	-
8.00	fpll-div	51	9615	25	19231	-	-	15	31250	12	38462
	fpll-div/4	12	9615	-	-	-	-	3	31250	-	-
	fpll-div/16	-	-	-	-	-	-	-	-	-	-
	fpll-div/32	-	-	-	-	-	-	-	-	-	-
	fpll-div/64	-	-	-	-	-	-	-	-	-	-
	fs/2	12	9615	-	-	-	-	3	31250	-	-
	fs/4	-	-	-	-	-	-	1	31250	-	-
8.38	fpll-div	54	9523	26	19398	-	-	-	-	-	-
	fpll-div/4	-	-	-	-	-	-	-	-	-	-
	fpll-div/16	-	-	-	-	-	-	-	-	-	-
	fpll-div/32	-	-	-	-	-	-	-	-	-	-
	fpll-div/64	-	-	-	-	-	-	-	-	-	-
	fs/2	-	-	-	-	-	-	-	-	-	-
	fs/4	-	-	-	-	-	-	-	-	-	-
12.00	fpll-div	77	9615	38	19231	25	28846	23	31250	-	-
	fpll-div/4	-	-	-	-	-	-	5	31250	-	-
	fpll-div/16	-	-	-	-	-	-	-	-	-	-
	fpll-div/32	-	-	-	-	-	-	-	-	-	-
	fpll-div/64	-	-	-	-	-	-	-	-	-	-
	fs/2	-	-	-	-	-	-	5	31250	-	-
	fs/4	-	-	-	-	-	-	2	31250	-	-
16.00	fpll-div	103	9615	51	19231	-	-	31	31250	25	38462
	fpll-div/4	25	9615	12	19231	-	-	7	31250	-	-
	fpll-div/16	-	-	-	-	-	-	-	-	-	-
	fpll-div/32	-	-	-	-	-	-	-	-	-	-
	fpll-div/64	-	-	-	-	-	-	-	-	-	-
	fs/2	25	9615	-	-	-	-	7	31250	-	-
	fs/4	12	9615	-	-	-	-	2	31250	-	-
20.00	fpll-div	129	9615	64	19231	-	-	39	31250	-	-
	fpll-div/4	-	-	-	-	-	-	9	31250	-	-
	fpll-div/16	-	-	-	-	-	-	-	-	-	-
	fpll-div/32	-	-	-	-	-	-	-	-	-	-
	fpll-div/64	-	-	-	-	-	-	-	-	-	-
	fs/2	-	-	-	-	-	-	9	31250	-	-
	fs/4	-	-	-	-	-	-	4	31250	-	-

Table:13.4.8 Setup Value of UART Serial Interface Transfer Speed (decimal) Clock Source Divided by 16

		Transfer speed (bps)									
fpll-div (MHz)	Clock source (timer)	300		960		1200		2400		4800	
		Set value	Calculated value	Set value	Calculated value	Set value	Calculated value	Set value	Calculated value	Set value	Calculated value
4.00	fpll-div	-	-	129	962	103	1202	51	2404	25	4808
	fpll-div/4	104	297	-	-	25	1202	12	2404	-	-
	fpll-div/16	25	300	-	-	-	-	-	-	-	-
	fpll-div/32	12	300	-	-	-	-	-	-	-	-
	fpll-div/64	-	-	-	-	-	-	-	-	-	-
	fs/2	103	300	-	-	25	1202	12	2404	-	-
	fs/4	51	297	-	-	12	1202	-	-	-	-
	fs/8	25	300	-	-	-	-	-	-	-	-
4.19	fpll-div	-	-	135	963	108	1201	-	-	-	-
	fpll-div/4	109	297	33	963	-	-	-	-	-	-
	fpll-div/16	26	303	-	-	-	-	-	-	-	-
	fpll-div/32	-	-	-	-	-	-	-	-	-	-
	fpll-div/64	-	-	-	-	-	-	-	-	-	-
	fs/2	109	297	33	963	-	-	-	-	-	-
	fs/4	54	297	16	963	-	-	6	2338	-	-
	fs/8	26	303	-	-	-	-	-	-	-	-
8.00	fpll-div	-	-	-	-	-	-	103	2404	51	4808
	fpll-div/4	208	300	64	962	51	1202	25	2404	12	4808
	fpll-div/16	51	300	-	-	12	1202	-	-	-	-
	fpll-div/32	25	300	-	-	-	-	-	-	-	-
	fpll-div/64	12	300	-	-	-	-	-	-	-	-
	fs/2	208	300	64	962	51	1202	25	2404	12	4808
	fs/4	103	300	-	-	25	1202	12	2404	-	-
	fs/8	51	300	-	-	12	1202	-	-	-	-
8.38	fpll-div	-	-	-	-	217	1202	108	2403	-	-
	fpll-div/4	218	300	67	963	-	-	27	2338	-	-
	fpll-div/16	55	297	16	963	-	-	6	2338	-	-
	fpll-div/32	27	303	-	-	-	-	-	-	-	-
	fpll-div/64	-	-	-	-	-	-	-	-	-	-
	fs/2	-	-	67	963	-	-	27	2338	-	-
	fs/4	109	300	33	963	-	-	13	2338	-	-
	fs/8	55	297	16	963	-	-	6	2338	-	-
12.00	fpll-div	-	-	-	-	-	-	155	2404	77	4808
	fpll-div/4	-	-	-	-	77	1202	38	2404	-	-
	fpll-div/16	77	300	-	-	-	-	-	-	-	-
	fpll-div/32	38	300	-	-	-	-	-	-	-	-
	fpll-div/64	-	-	-	-	-	-	-	-	-	-
	fs/2	-	-	-	-	77	1202	38	2404	-	-
	fs/4	154	302	-	-	38	1202	-	-	-	-
	fs/8	77	300	-	-	-	-	-	-	-	-
16.00	fpll-div	-	-	-	-	-	-	207	2404	103	4808
	fpll-div/4	-	-	-	-	103	1202	51	2404	25	4808
	fpll-div/16	103	300	-	-	25	1202	12	2404	-	-
	fpll-div/32	51	300	-	-	12	1202	-	-	-	-
	fpll-div/64	25	300	-	-	-	-	-	-	-	-
	fs/2	-	-	-	-	103	1202	51	2404	25	4808
	fs/4	208	300	64	962	51	1202	25	2404	12	4808
	fs/8	103	300	-	-	25	1202	12	2404	-	-
20.00	fpll-div	-	-	-	-	-	-	-	-	-	-
	fpll-div/4	-	-	162	959	129	1201	64	2403	-	-
	fpll-div/16	129	300	-	-	-	-	-	-	-	-
	fpll-div/32	64	300	-	-	-	-	-	-	-	-
	fpll-div/64	-	-	-	-	-	-	-	-	-	-
	fs/2	-	-	162	959	129	1201	64	2403	-	-
	fs/4	129	300	80	965	64	1201	-	-	-	-
	fs/8	129	300	-	-	-	-	-	-	-	-

Table:13.4.9 Setup Value of UART Serial Interface Transfer Speed (decimal) Clock Source Divided by 16

		Transfer speed (bps)									
fpll-div (MHz)	Clock source (timer)	9600		19200		28800		31250		38400	
		Set value	Calculated value	Set value	Calculated value	Set value	Calculated value	Set value	Calculated value	Set value	Calculated value
4.00	fpll-div	12	9615	-	-	-	-	3	31250	-	-
	fpll-div/4	-	-	-	-	-	-	-	-	-	-
	fpll-div/16	-	-	-	-	-	-	-	-	-	-
	fpll-div/32	-	-	-	-	-	-	-	-	-	-
	fpll-div/64	-	-	-	-	-	-	-	-	-	-
	fs/2	-	-	-	-	-	-	-	-	-	-
	fs/4	-	-	-	-	-	-	-	-	-	-
	fs/8	-	-	-	-	-	-	-	-	-	-
4.19	fpll-div	-	-	-	-	-	-	-	-	-	-
	fpll-div/4	-	-	-	-	-	-	-	-	-	-
	fpll-div/16	-	-	-	-	-	-	-	-	-	-
	fpll-div/32	-	-	-	-	-	-	-	-	-	-
	fpll-div/64	-	-	-	-	-	-	-	-	-	-
	fs/2	-	-	-	-	-	-	-	-	-	-
	fs/4	-	-	-	-	-	-	-	-	-	-
	fs/8	-	-	-	-	-	-	-	-	-	-
8.00	fpll-div	25	9615	12	19231	-	-	7	31250	-	-
	fpll-div/4	-	-	-	-	-	-	1	31250	-	-
	fpll-div/16	-	-	-	-	-	-	-	-	-	-
	fpll-div/32	-	-	-	-	-	-	-	-	-	-
	fpll-div/64	-	-	-	-	-	-	-	-	-	-
	fs/2	-	-	-	-	-	-	1	31250	-	-
	fs/4	-	-	-	-	-	-	-	-	-	-
	fs/8	-	-	-	-	-	-	-	-	-	-
8.38	fpll-div	26	9699	-	-	-	-	-	-	-	-
	fpll-div/4	-	-	-	-	-	-	-	-	-	-
	fpll-div/16	-	-	-	-	-	-	-	-	-	-
	fpll-div/32	-	-	-	-	-	-	-	-	-	-
	fpll-div/64	-	-	-	-	-	-	-	-	-	-
	fs/2	-	-	-	-	-	-	-	-	-	-
	fs/4	-	-	-	-	-	-	-	-	-	-
	fs/8	-	-	-	-	-	-	-	-	-	-
12.00	fpll-div	38	9615	-	-	12	28847	11	31251	-	-
	fpll-div/4	-	-	-	-	-	-	2	31251	-	-
	fpll-div/16	-	-	-	-	-	-	-	-	-	-
	fpll-div/32	-	-	-	-	-	-	-	-	-	-
	fpll-div/64	-	-	-	-	-	-	-	-	-	-
	fs/2	-	-	-	-	-	-	2	31250	-	-
	fs/4	-	-	-	-	-	-	-	-	-	-
	fs/8	-	-	-	-	-	-	-	-	-	-
16.00	fpll-div	51	9615	25	19231	-	-	11	31250	12	38462
	fpll-div/4	12	9615	-	-	-	-	3	31250	-	-
	fpll-div/16	-	-	-	-	-	-	-	-	-	-
	fpll-div/32	-	-	-	-	-	-	-	-	-	-
	fpll-div/64	-	-	-	-	-	-	-	-	-	-
	fs/2	12	9615	-	-	-	-	3	31250	-	-
	fs/4	-	-	-	-	-	-	1	31250	-	-
	fs/8	-	-	-	-	-	-	-	-	-	-
20.00	fpll-div	64	9615	-	-	-	-	19	31250	-	-
	fpll-div/4	-	-	-	-	-	-	4	31250	-	-
	fpll-div/16	-	-	-	-	-	-	-	-	-	-
	fpll-div/32	-	-	-	-	-	-	-	-	-	-
	fpll-div/64	-	-	-	-	-	-	-	-	-	-
	fs/2	-	-	-	-	-	-	4	31250	-	-
	fs/4	-	-	-	-	-	-	-	-	-	-
	fs/8	-	-	-	-	-	-	-	-	-	-

The items shown below are the same as clock synchronous serial. Refer to the following pages.

- First Transfer Bit Setup

Refer to: XIII-38

- Transmission Data Buffer

Refer to: XIII-38

- Reception Data Buffer

Refer to: XIII-39

- Transmit Bit Count and First Transfer Bit

Refer to: XIII-39

- Transmission Buffer Empty Flag

Refer to: XIII-43

- Emergency Reset

Refer to: XIII-44

13.4.2 Timing

■ Transmission Timing

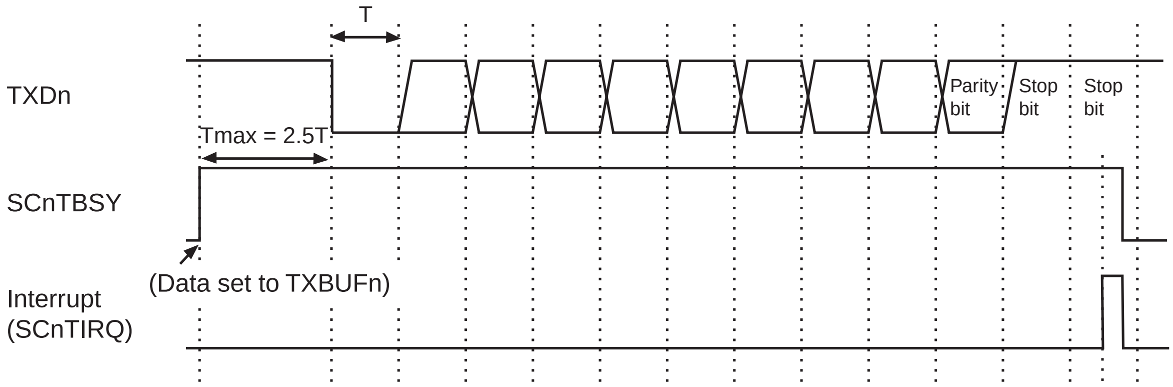


Figure:13.4.4 Transmission Timing (Parity Bit is Enabled)

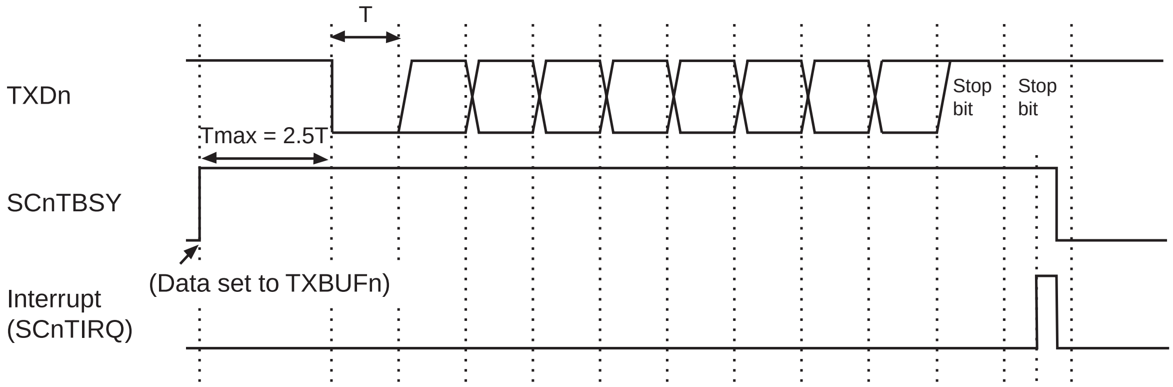


Figure:13.4.5 Transmission Timing (Parity Bit is Disabled)

■ Reception Timing

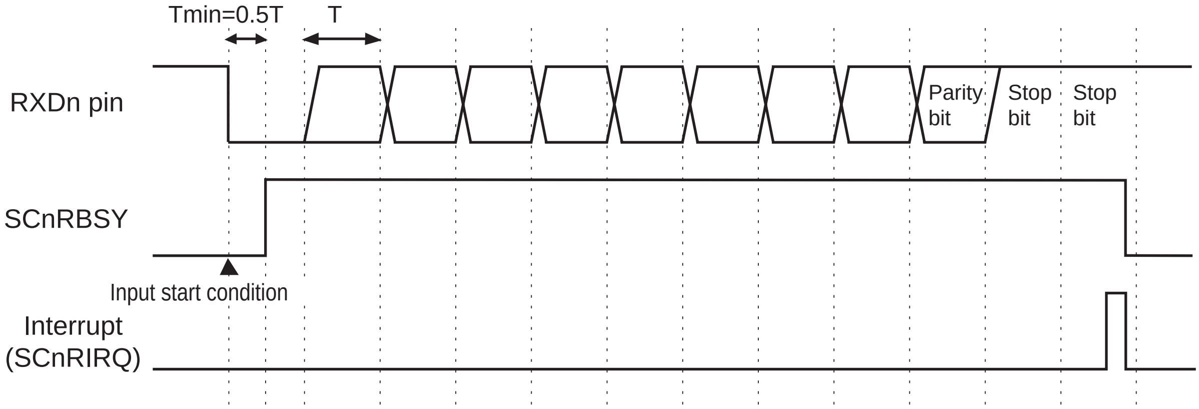


Figure:13.4.6 Reception Timing (Parity Bit is Enabled)

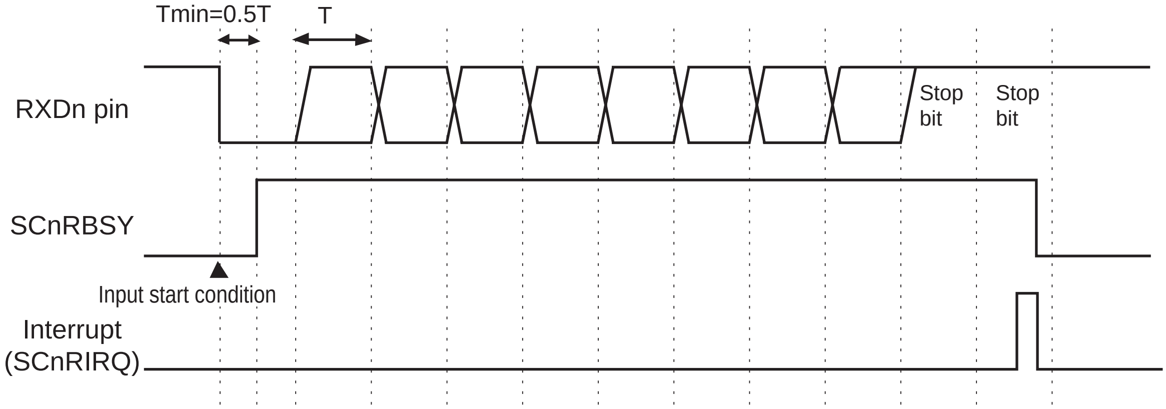


Figure:13.4.7 Reception Timing (Parity Bit is Disabled)

13.4.3 Pin Setup

■ UART Serial Interface 0 Pin Setup

Port	Wire system	Type	Pin	Pin setup (flag setup)						
				SC0SEL register	PnDIR register	PnODC register	PnPLUD register	SC0MD1 register		
				Serial 0 I/O pin switching	I/O mode selection	Nch open-drain output selection Arbitrary setting	Pull-up /Pull-down resistor selection Aribitary setting	Serial data input selection	Serial input control selection	SBO0 (RXD0) pin function selection
				0:A type 1:B type	0: input mode 1: output mode	0: push/ pull 1: Nch open-drain	0: not added 1: added	0: data input from RXD0 1: data input from TXD0	0:"1" input 1: serial input	0: port 1: serial data output
				0SL0	PnDIRm	PnODCm	PnPLUm	SC0IOM	SC0SBIS	SC0SBOS
Port 0	2-wire	Transmission only	P03/TXD0A	0	P0DIR3:1	P0ODC3	P0PLU3	0	0	1
			-		-	-	-			
		Reception only	-		-	-	-	0	1	0
			P02/RXD0A		P0DIR2:0	-	-			
		Transmission/ Reception	P03/TXD0A		P0DIR3:1	P0ODC3	P0PLU3	0	1	1
			P02/RXD0A		P0DIR2:0	-	-			
	1-wire	Transmission	P03/TXD0A		P0DIR3:1	P0ODC3	P0PLU3	0	0	1
			-		-	-	-			
		Reception	P03/TXD0A		P0DIR3:0	-	-	1	1	0
			-		-	-	-			
Port 4	2-wire	Transmission only	P43/TXD0B	1	P4DIR3:1	P4ODC3	P4PLU3	0	0	1
			-		-	-	-			
		Reception only	-		-	-	-	0	1	0
			P44/RXD0B		P4DIR4:0	-	-			
		Transmission/ Reception	P43/TXD0B		P4DIR3:1	P4ODC3	P4PLU3	0	1	1
			P44/RXD0B		P4DIR4:0	-	-			
	1-wire	Transmission	P43/TXD0B		P4DIR3:1	P4ODC3	P4PLU3	0	0	1
			-		-	-	-			
		Reception	P43/TXD0B		P4DIR3:0	-	-	1	1	0
			-		-	-	-			

■ UART Serial Interface 1 Pin Setup

Port	Wire system	Type	Pin	Pin setup (flag setup)						
				SC1SEL register	PnDIR register	PnODC register	PnPLUD register	SC1MD1 register		
				Serial 1 I/O pin switching	I/O mode selection	Nch open-drain output selection Arbitrary setting	Pull-up /Pull-down resistor selection Aribitary setting	Serial data input selection	Serial input control selection	SBO1 (RXD1) pin function selection
				0:A type 1:B type	0: input mode 1: output mode	0: push/ pull 1: Nch open-drain	0: not added 1: added	0: data input from SBI1 1: data input from SBO1	0:"1" input 1: serial input	0: port 1: serial data output
				0SL1	PnDIRm	PnODCm	PnPLUm	SC1IOM	SC1SBIS	SC1SBOS
Port 5	2-wire	Transmission only	P50/TXD1A	0	P5DIR0:1	P5ODC0	P5PLU0	0	0	1
			-		-	-	-			
		Reception only	-		-	-	-	0	1	0
			P51/RXD1A		P5DIR1:0	-	-			
		Transmission/Reception	P50/TXD1A		P5DIR0:1	P5ODC0	P5PLU0	0	1	1
			P51/RXD1A		P5DIR1:0	-	-			
	1-wire	Transmission	P50/TXD1A		P5DIR0:1	P5ODC0	P5PLU0	0	0	1
			-		-	-	-			
		Reception	P50/TXD1A		P5DIR0:0	-	-	1	1	0
			-		-	-	-			
Port 7	2-wire	Transmission only	P75/TXD1B	1	P7DIR5:1	P7ODC5	P7PLU5	0	0	1
			-		-	-	-			
		Reception only	-		-	-	-	0	1	0
			P76/RXD1B		P7DIR6:0	-	-			
		Transmission/Reception	P75/TXD1B		P7DIR5:1	P7ODC5	P7PLU5	0	1	1
			P76/RXD1B		P7DIR6:0	-	-			
	1-wire	Transmission	P75/TXD1B		P7DIR5:1	P7ODC5	P7PLU5	0	0	1
			-		-	-	-			
		Reception	P75/TXD1B		P7DIR5:0	-	-	1	1	0
			-		-	-	-			

■ UART Serial Interface 2 Pin Setup

Port	Wire system	Type	Pin	Pin setup (flag setup)					
				PnDIR register	PnODC register	PnPLUD register	SC2MD1 register		
				I/O mode selection	Nch open-drain output selection Arbitrary setting	Pull-up /Pull-down resistor selection Aribitrary setting	Serial data input selection	Serial input control selection	SBO2 (RXD2) pin function selection
				0: input mode 1: output mode	0: push/ pull 1: Nch open-drain	0: not added 1: added	0: data input from SBI2 1: data input from SBO2	0:"1" input 1: serial input	0: port 1: serial data output
				PnDIRm	PnODCm	PnPLUm	SC2IOM	SC2SBIS	SC2SBOS
Port 6	2-wire	Transmission only	P65/TXD2	P6DIR5:1	P6ODC5	P6PLU5	0	0	1
			-	-	-	-			
		Reception only	-	-	-	-	0	1	0
			P66/RXD2	P6DIR6:0	-	-			
		Transmission/ Reception	P65/TXD2	P6DIR5:1	P6ODC5	P6PLU5	0	1	1
			P66/RXD2	P6DIR6:0	-	-			
	1-wire	Transmission	P65/TXD2	P6DIR5:1	P6ODC5	P6PLU5	0	0	1
			-	-	-	-			
		Reception	P65/TXD2	P6DIR5:0	-	-	1	1	0
			-	-	-	-			

13.4.4 Setup Example

■ Transmission/Reception Setup

The setup example at UART transmission/reception using Serial Interface 0 is shown below. Table:13.4.10 shows the conditions at transmission/reception. The basic procedures are the same in Serial Interface 0, 1 and 2. Pin settings (2) and (3) differ in each serial interface.

Table:13.4.10 UART Interface Transmission Reception Setup Condition

Setup item	Set to
Serial data input selection	RXD0
Frame mode specification	8 bits + 2 stop bits
First transfer bit	MSB
Clock source	Timer A
TXD0/RXD0 pin style	Nch open-drain
Pull-up resistor of TXD0 pin	Added
Parity bit add/check	"0" added/checked
Serial 0 transmission complete interrupt	Enabled
Serial 0 reception complete interrupt	Enabled

An example setup procedure, with a description of each step is shown below

Setup Procedure	Description
(1) Select the clock source SC0SEL (0x03F10) bp2 to 0: SC0SEL2 to 0 =111 SC0MD3 (0x03F10) bp2 to 0: SC0PSC2 to 0 =111	(1) Set the SC0SEL2 to 0 flags of SC0SEL register and the SC0PSC2 to 0 flags of SC0MD3 register to "111" to select Timer A output as a clock source.
(2) Control the pin style [set the pin corresponding to each serial] P0ODC(0x03EF0) bp3: P0ODC3 =1 P0PLU(0x03FA0) bp3: P0PLU3 =1	(2) Set the P0ODC3 flags of P0ODC register to "1" to select Nch open-drain as styles of TXD1 pin. Set the P0PLU3 flag of P0PLU register to "1" to enable the pull to up resistor. (Set the pin corresponding to each serial interface.)
(3) Control the pin direction [set the pin corresponding to each serial] P0DIR(0x03F90) bp3 to 2: P0DIR3 to 2 = 10	(3) Set the P0DIR3 to 2 flags of P0DIR register to "10" to set P02 to input mode and P03 to output mode. (Set the pin corresponding to each serial interface.)
(4) Set SC0MD0 register Select the transfer first bit SC0MD0 (0x03F11) bp4: SC0DIR =0	(4) Set the SC0DIR flag of SC0MD0 register to "0" to select MSB as the transfer first bit.

Setup Procedure	Description
(5) Set SC0MD2 register Control the output data SC0MD2 (0x03F13) bp0: SC0BRKE =0 Select the added parity bit SC0MD2 (0x03F13) bp3: SC0NPE =0 bp5 to 4: SC0PM1 to 0 =00 Specify the frame mode SC0MD2 (0x03F13) bp7 to 6: SC0FM1 to 0 =11	(5) Set the SC0BRKE flag of SC0MD2 register to "0" to select serial data transmission. Set the SC0PM1 to 0 flags of SC0MD2 register to "00" to select 0 parity, and set the SC0NPE flag to "0" to enable add parity bit. Set the SC0FM1 to 0 flags of SC0MD2 register to "11" to select 8 bits + 2 stop bits at the frame mode.
(6) Set SC0MD1 register Select the communication type SC0MD1 (0x03F12) bp0: SC0CMD =1 Select the clock dividing SC0MD1 (0x03F12) bp3: SC0CKM =1 bp2: SC0MST =1 bp1: SC0DIV =0 Control the pin function SC0MD1 (0x03F12) bp4: SC0SBOS =1 bp5: SC0SBIS =1 bp6: SC0SBTS =0 bp7: SC0IOM =0	(6) Set the SC0CMD flag of SC0MD1 register to "1" to select the duplex UART. Set the SC0CKM flag of SC0MD1 register to "1" to select “Divided” for transfer clock division selection. Set the SC0DIV flag of SC0MD1 register to “0” to select "Divide by 8" for transfer clock division value selection. The SC0MST flag should be always set to "1" to select a clock master. Set the SC0SBOS and SC0SBIS flags of SC0MD1 register to "1" and the SC0SBTS flag of SC0MD1 register to "0" to set TXD0 pin to serial data output and RXD0 pin to serial data input and SBT0 pin to port.
(7) Set the baud rate timer	(7) Set the baud rate by TMAMD register and TMAOC register. Set the TMAEN flag to "1" to operate Timer A. [Chapter 7 7.4.1 Operation]
(8) Enable the interrupt PSW bp6: MIE =0 SC0RICR(0x03FFC) bp1: SC0RIE =1 SC0TICR(0x03FFB) bp1: SC0TIE =1 PSW bp6: MIE =1	(8) Clear the MIE flag of PSW to disable all maskable interrupts. Set the SC0RIE flag of SC0RICR register to "1" and also set the SC0TIE flag of SC0TICR register to "1" to enable the interrupt request. If any of the interrupt request is already set, clear the request flag. Set the MIE flag of PSW to "1" to enable maskable interrupt.
(9) Start the serial transmission The transmission data → TXBUF0 (0x03F17) The reception data → input to RXD0 pin	(9) Transmission is started by setting transmission data to the serial transmission data buffer (TXBUF0). When the transmission is completed, Serial 0 transmission complete interrupt (SC0TIRQ) is generated. If a start bit is detected, the received data is loaded into the RXBUF0; and, Serial 0 reception complete interrupt (SC0RIRQ) is generated.

* Each setup in (4), (5) and (6) can be set at the same time.

For 1-wire communication, input and output serial data from TXDn pin. Use port direction control register to switch between input and output. In reception, be sure to set SCnSBIS of SCnMD1 register to "1" to select "serial data input". RXDn pin can be used as a general purpose port.

This serial interface has an emergency reset function. If communication needs to be stopped by this function, set SCnSBOS and SCnSBIS of SCnMD1 register to "0".

Set each flag in accordance with the procedures indicated in the table above. Be sure to start communication after the settings of all control registers (refer to Table:13.2.1 but exclude TXBUFn and RXBUFn) are completed.

When setting this serial interface communication mode to “UART”, set Serial Interface n mode register 1 (SCnMD1) while the serial data input pin is at “High”.

When the SCnCMD flag of SCnMD1 register is set to “1” to select “Duplex UART”, the transfer clock is divided regardless of the SCnCKM flag.

13.5 Multi Master IIC Interface

13.5.1 Multi Master IIC Interface

Multi master IIC serial communication is available with serial interface 4. This IIC interface communicates by complying with the data transfer format of Philips IIC-BUS. Table:13.1.6 shows IIC serial interface functions.

■ Data I/O Pin Setup

Use SDA4 pin for data input/output, which is commonly used with SBO4 pin. Set the SC4IOM flag of SC4MD1 register to "1" to input the serial data from SDA4 pin. SBI4 pin can be used as a general-purpose port; make sure to set the SC4SBIS flag of SC4MD1 register to "1" to set "serial data input".

Be sure to set the SC4SBIS flag of SC4MD1 register to "serial data input" regardless of transmission/reception in order to detect the start condition and ACK bit reception.

Nch open-drain should be used for pin format because the bus is switched between use and open by hardware during communication. Even in reception, select "output" for direction control of SDA pin.

■ Input Edge/Output Edge Setup

In IIC communication, data is always received at the falling edge of the clock regardless of the SC4CE1 value.

■ Master/Slave Selection

Multi master IIC function (clock master/clock slave) or slave-dedicated IIC function (clock slave) can be selected with the SC4MST of SC4MD1 register. When a start condition from another master or arbitration lost is detected if using clock master, clock slave operation will be selected.

■ Slave Address Setup

In this serial interface, 7-bit or 10-bit slave address can be set. To change slave address for 7 bits, set slave address to the upper 7 bits of SC4AD0 register (SC4ADR7 to SC4ADR1) after setting the SC4ADM flag of SC4MD3 register to "0" to set the 7-bit address mode. To change slave address for 10 bits, set the upper 2 bits of slave address to the lower 10 bits of SC4AD1 register (SC4ADR9 and SC4ADR8), and set the lower 8 bits of slave address to SC4AD0 register after setting the SC4ADM flag of SC4MD3 register to "1" to set the 10-bit address mode.

■ Transfer Format

Two formats are available for IIC bus transfer; addressing format and free data format. In addressing format, 1 to 2 byte address data which consists of the slave address (7 bits/10 bits) and R/W bit (1 bit) is transmitted after the start condition, and transmission/reception is executed. In free data format, data is transmitted immediately after the start condition. Regarding the free data format, this LSI only supports IIC master communication. The following presents the communication sequences. The shaded regions of Figure:13.5.1, Figure:13.5.3 and Figure:13.5.3 are the data transmitted from other IIC.

■ Addressing Format

- 7-bit address

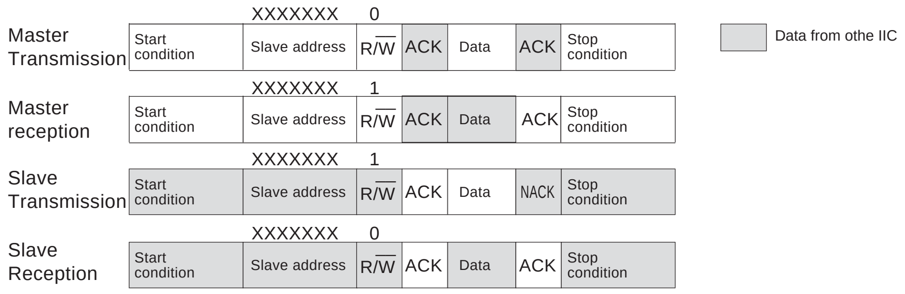


Figure:13.5.1 Communication Sequence in 7-bit Address Mode

- 10-bit address

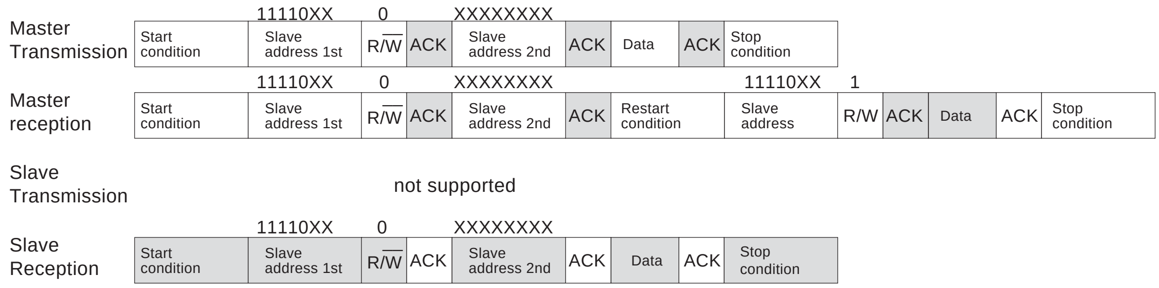


Figure:13.5.2 Communication Sequence in 10-bit Address Mode

- Free data format

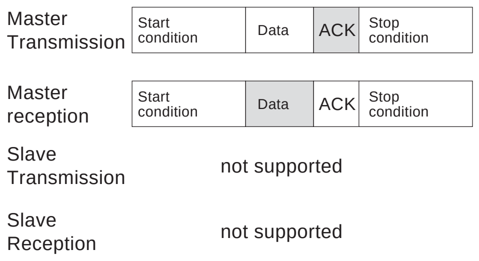


Figure:13.5.3 Communication Sequence in Free Data Format

■ Clock Setup

The clock source is selected from the dedicated prescaler and timer output (Timer 0 to 3 or A) by SC4MD2 register. The dedicated prescaler starts to operate by selecting "prescaler count enable" by the SC4MD2.

The following table shows the clock source selection condition.

Table:13.5.1 Clock Source Selection Condition

Communication condition	Clock source selection condition	
Master communication only (Only this IIC is set as the master)	Standard mode	800 kHz or less
	High-speed mode	3.2 MHz or less
Master/slave communication	Standard mode	500 kHz or more, 800 kHz or less
	High-speed mode	3.2 MHz
Slave communication only (SC4MST=0)	Standard mode	500 kHz or more
	High-speed mode	3.4 MHz or more

In master communication, the transfer clock is obtained by dividing the clock source by 8. Duty is "High:Low=1:1" in standard mode and "High:Low=3:5" in high-speed mode. In master communication, set the clock source so that the transfer clock is not over 100 kHz in standard mode and 400 kHz in high-speed mode. Select the transfer rate so that its value is 0.8 times or over of other master.

In slave communication, Set the clock source to 500 kHz or over in standard mode and 3.4 MHz or over in high-speed mode. However, when the transfer rate and duty of the master are identified, any clock source which satisfies the following conditions can be selected.

Table:13.5.2 Transfer Rate Duty and Clock Source Selection Condition

Transfer rate duty	Slave clock source selection condition
1:1	Transfer rate × 4 or more
1:2/2:1	Transfer rate × 6 or more
1:3/3:1	Transfer rate × 8 or more

Table:13.5.3 IIC Interface Clock Source

	Multi master IIC
Clock source (Internal clock)	fpll-div/2
	fpll-div/4
	fpll-div/16
	fpll-div/32
	fs/2
	fs/4
	Timer output (0 to 3 or A)

In IIC master communication, transfer rate is obtained by dividing the clock source by 8. In master communication, set the clock source with SC4MD2 register so that the transfer rate is not over 100 kHz in the standard mode and 400 kHz in the high-speed mode.
Select the transfer rate which is more than 0.8 times of the other master.

In slave communication, set the clock source to 500 kHz or more in the standard mode, 3.4 MHz or more in the high-speed mode, or set to the value which is 6 times or larger than the transfer rate.

When switching the clock setup, always set the SC4SBIS flag and SC4SBOS flag of SC4MD1 register to "0" in advance.

■ Activation Factor for Communication

(Master communication)

Set data (at transmission) or dummy data (at reception) in the transmission buffer (TXBUF4). Regardless of transmission or reception, a start condition and transfer clock are generated to start communication.

(Slave communication)

Detecting a start condition starts reception. When the received address matches the address set by the address setting registers 0 and 1, or a general call is detected, the slave address comparison flag is set to start slave communication.

■ Interrupt

This serial interface has two types of interrupt, a Serial 4 interrupt "SC4IRQ" and a Serial 4 stop condition detection interrupt "SC4STPCIRQ". Table:13.5.4 shows the interrupt generation factors.

Table:13.5.4 IIC Communication Interrupt Generation Factor

Interrupt	Interrupt generation factor
SC4IRQ (Serial 4 interrupt)	Master communication completion (after 1 byte data + ACK)
	Slave address match (after ACK)
	Slave communication completion (after 1 byte data + ACK)
	Slave communication completion (communication data instability detection)
SC4STPCIRQ (Serial 4 stop condition detection interrupt)	Detection of stop condition generated by other master

However, in 10-bit address mode communication, even if the first 2 bits addresses transmitted from master match with SC4ADR9 to 8 flags of SC4AD1 register, Serial 4 communication completion interrupt (SC4IRQ) is not generated. Only when the next lower 8 bits addresses match with SC4ADR7 to 0 flags of SC4AD0 register, SC4IRQ is generated.

■ Start Condition Setup

In master communication, a start condition is always generated at the first communication regardless of the SC4STE flag value of SC4MD0 register. Select whether the start condition is enabled or disabled after the second byte communication by the SC4STE flag of SC4MD0 register.

Enable/disable of the start condition can be determined in each communication by setting the enable/disable before each communication data setup.

Start Condition Setup Example

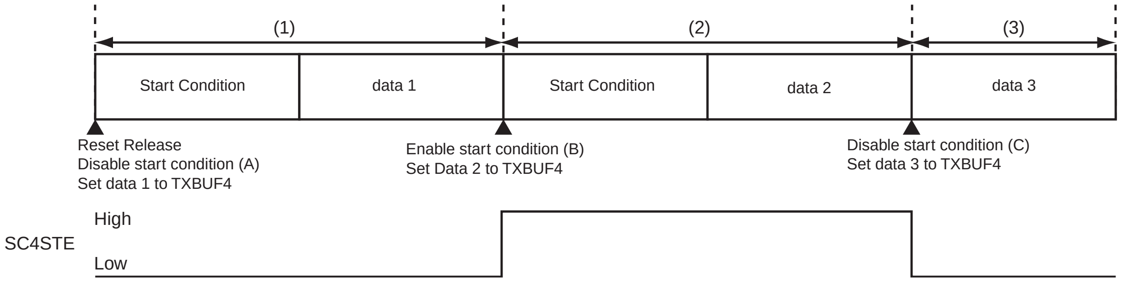


Figure:13.5.4 Start Condition Setup Example

(1) At the first communication, start condition is added regardless of SC4STE value.

(2) Start condition is added by setting start condition to "enable" at (B).

(3) Start condition is not added by setting start condition to "disable" at (C).

■ Stop Condition Generation

Stop condition is formed if the data bus (SDA4 pin) changes from "Low" to "High" when the clock bus (SCL4 pin) is "High". Writing "1" in the SC4STPC flag of SC4MD3 register by program starts the stop condition output. When the stop condition is generated, the SC4STPC flag is automatically cleared.

The stop condition should be requested only when this IIC occupies the bus as master. If the stop condition is generated by this IIC, Serial 4 stop condition detection interrupt is not generated.

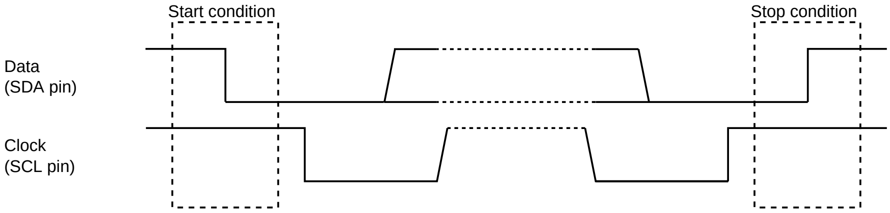


Figure:13.5.5 Start Condition and Stop Condition

■ Start/Restart Condition Detection

If the data bus (SDA4 pin) changes from "High" to "Low" when the clock bus (SCL4 pin) is "High", a start condition is detected and the SC4STRT flag and SC4BUSBSY flag of SC4STR1 register are set to "1". The SC4STRT flag is cleared to "0" when data is set in TXBUF4 by the interrupt process immediately after the slave address reception. The SC4STRT flag is also set when a restart condition is detected. If the address transmitted from master does not match with the slave address, the address is automatically cleared by the hardware when an address mismatch is detected.

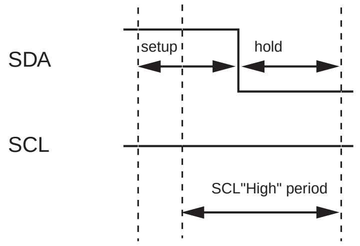
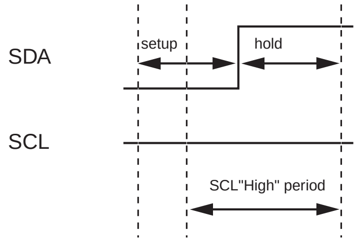
■ Stop Condition Detection

If the data bus (SDA4 pin) changes from "Low" to "High" while the clock bus (SCL4 pin) is "High", a start condition is detected. SC4STPCIRQ is generated and the SC4BUSBSY flag of SC4STR1 register is cleared.

■ Start Condition/Stop Condition Detection Condition

This IIC detects the start condition and the stop condition on bus lines by the sampling with the internal clock. The detection conditions are as follows.

Table:13.5.5 Start Condition/Stop Condition Detection Condition

Start Condition		SCL "High" period	3-clock source or more
		SDA setup	2-clock source or more
		SDA hold	2-clock source or more
Stop Condition		SCL "High" period	3-clock source or more
		SDA setup	2-clock source or more
		SDA hold	2-clock source or more

■ Communication Data Instability Detection

When a serial circuit detects a changes in the data bus (SDA4 pin) while the clock bus (SCL4 pin) is "High" after a start condition is detected, the communication data is recognized as instability and the SC4DATA_ERR flag of SC4STR1 register is set to "1". Since communication is not properly executed when the SC4DATA_ERR flag is "1", clear the SC4DATA_ERR flag to "0" by a user's program and start the communication again.
The first operation after communication data instability is detected is different between the master communication and the slave communication.
In the case of the master communication, the communication completion interrupt SC4IRQ is generated after completing 1 byte communication. However, if the arbitration lost is detected, the communication mode is changed from the master to the slave, and the communication may be stopped.
On the other hand, in slave communication, the SC4IRQ is generated immediately and the communication is completed. When the SC4DEM flag is "1" in slave communication, the circuit can detect start conditions and start communication even if the SC4DATA_ERR flag is "1".

■ Transmission at Master Communication/Reception Mode Setup and Operation

In master communication, select the transmission or reception mode by the SC4REX flag of SC4MD3 register. The first data always communicates with a start condition regardless of the value set to the SC4STE flag. The start condition is output from this master serial interface.

In master operation, transmission is executed by setting the slave address and R/W bit in the first data after a start condition is generated during the addressing format. In the master reception, check the ACK signal from the slave in the interrupt process after the address data transmission is finished, then switch to the reception mode.

To start a new communication with other devices before the current communication end, regenerate a start condition and transmit the slave address and R/W bit again. In reception, SDA4 line is automatically released and will be in reception wait status. The reception acknowledge (ACK bit) is transmitted after the data storage is completed.
Refer to [Figure:13.5.12 Master Transmission Timing] and [Figure:13.5.13 Master Reception Timing].

■ Slave Communication

This serial interface performs address verification by obtaining the received data automatically after detecting a start condition on IIC bus. SC4IRQ is generated only when the address transmitted from master matches with the assigned slave address. Data transmission and reception are determined by the SC4WRS flag of SC4STR1 register. When SC4WRS = "0", slave reception is selected; when SC4WRS = "1", slave transmission is selected. In slave transmission, the bus line is released by setting the transmission data in TXBUF4 register, and data transmission starts with the clock transmitted from the master. Do not set data to TXBUF4 register because bus line is released automatically when NACK is received. In slave reception, the bus line is released by setting dummy data to TXBUF4 register, and data reception starts with the clock transmitted from the master.

■ Address Compare Flag

When the address transmitted from master matches with the slave address, the address comparison flag SC2ADD_ACC of SC2STR1 is set to "1" and ACK is output automatically. In 10-bit address mode communication, when the first upper 2 bits of addresses transmitted from master match with SC2AD9 to 8 flags of SC2AD1, the SC2ADD_ACC flag is set to "1" and ACK is output automatically. When the next lower 8 bits of addresses transmitted from master match with SC2AD7 to 0 flags of SC2AD0 register, the SC2ADD_ACC flag is held at "1" and ACK bit is output automatically. When the addresses do not match, the SC2ADD_ACC flag is cleared to "0" and also ACK will not be output.

■ General Call Communication outputted

This serial interface supports general call communication. When a general call is detected, the SC4ADD_ACC flag and SC4GCALL flag of SC4STR1 register are set.

■ Reception of Acknowledgement (ACK bit) after Data Transmission

ACK bit can be enabled or disabled by the SC4ACKS flag of SC4MD3 register. If ACK bit is enabled, data (2 to 8 bits) is transmitted and ACK bit is received from the data receiver. The data bus (SDA) is released automatically upon receiving the ACK bit. In master operation, the clock for ACK bit reception is output once, and the ACK bit is stored in the SC4ACK0 flag of SC4MD3 register. There is no shift operation for reception register RXBUF4 by the ACK bit reception clock. When the received ACK bit level is "Low", the reception of the receiver is operated properly. It indicates the reception wait status for the next data.

When the ACK bit level is "High", the receiver may finish the reception process. At master operation, finish the communication by writing "1" to the SC4STPC flag of SC4MD3 register or issue the slave address again by generating a restart condition. At slave operation, it is not necessary to set data to TXBUF4 register because the transmission is finished by releasing the data bus (SDA4) automatically at slave operation. In this case, the slave address compare flag is cleared. To continue the communication, an address match is required again.

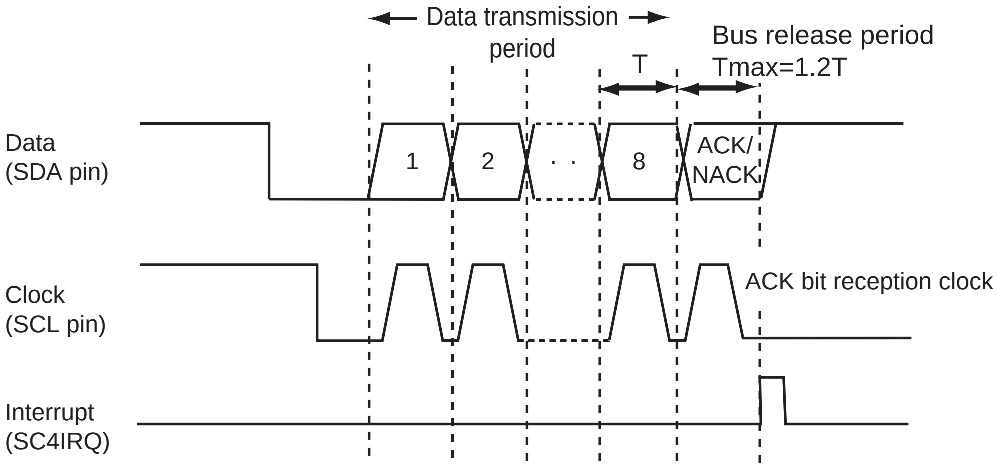


Figure:13.5.6 ACK Bit Reception Timing after 8-bit Data Transmission

■ Transmission of Acknowledgement (ACK bit) after Data Reception

Enable/disable of ACK bit can be selected in the same way as the ACK bit reception. When ACK bit is enabled, ACK bit and clock is output after receiving the data (1 to 8 bit). To continue the reception, output a "Low" level ACK bit. To finish the reception, output a "High" level ACK bit. ACK bit output level can be set by SC4ACK0 flag of SC4MD3 register.

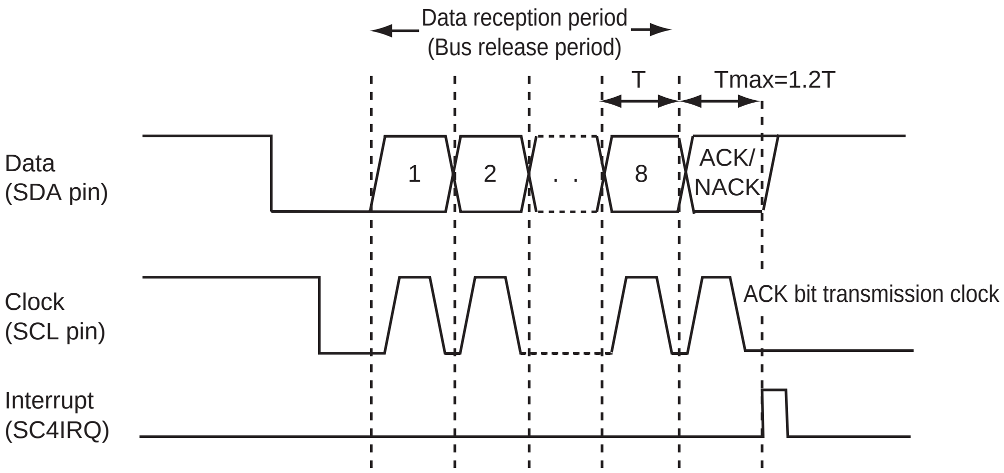


Figure:13.5.7 ACK Bit Transmission Timing after 8-bit Data Reception

■ Arbitration Lost

During master transmission, data bus (SDA) compares every bit of data output from this circuit to detect a race condition with other masters. When the output data does not match with the data bus, it judges as the communication is not permitted (arbitration lost). The SC4ABT_LST flag of SC4STR1 register is set instantly. It releases the data bus and the clock bus and continues slave reception. SC4IRQ is not generated when the arbitration lost is detected. However, SC4IRQ is generated when the slave address is matched after the detection of arbitration lost. Confirm the SC4ABT_LST flag at next interrupt generation timing (SC4STPCIRQ, SC4IRQ). When SC4ABT_LST flag is "1", execute communication again after the release of IIC bus because the master transmission is not formed.
In this case, clear the SC4ABT_LST flag with program.

■ Busy Flag

This serial interface contains 2 types of busy flags (SC4BUSBSY, SC4IICBSY) in SC4STR1 register.

The SC4BUSBSY flag is set to "1" in IIC bus communication. It is automatically set when a start condition is detected on IIC bus and cleared when a stop condition is detected.

The SC4IICBSY flag is "1" when this serial interface is in communication. In master communication, this flag is "1" during data loading, start condition generation, data communication, ACK communication, and stop condition generation. In slave communication, the flag is "1" ACK communication, data communication, and ACK output when the address transmitted from the master matches the slave address. If a restart condition is detected, the SC4IICBSY flag is cleared, and it becomes "1" again at ACK output when the address transmitted from the master matches the slave address. Refer to the timing charts described in [Figure:13.5.12 Master Transmission Timing] for the timing of flag set/clear.

A maximum of one cycle of internal transfer clock is needed before the communication starts after data is set to TXBUF4 register and the flag setting of SC4IICBY is done.

■ Handshake Using Clock Synchronous Mechanism

In master operation, SCL bus-line is monitored by sampling with the clock source. When different signal level between SCL output and SCL bus-line is detected, it is judged that the handshake with clock synchronous mechanism is activated, and the transfer clock is extended. With this operation, the master speed can be matched to the bus clock speed.

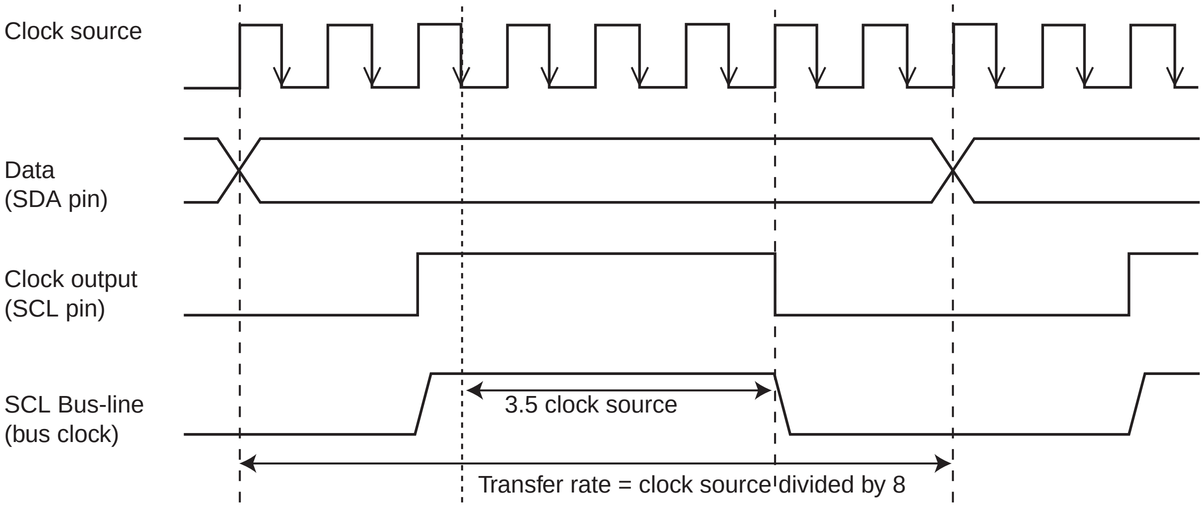


Figure:13.5.8 Without "Low" Period Extension from the Other Clock (Standard Mode)

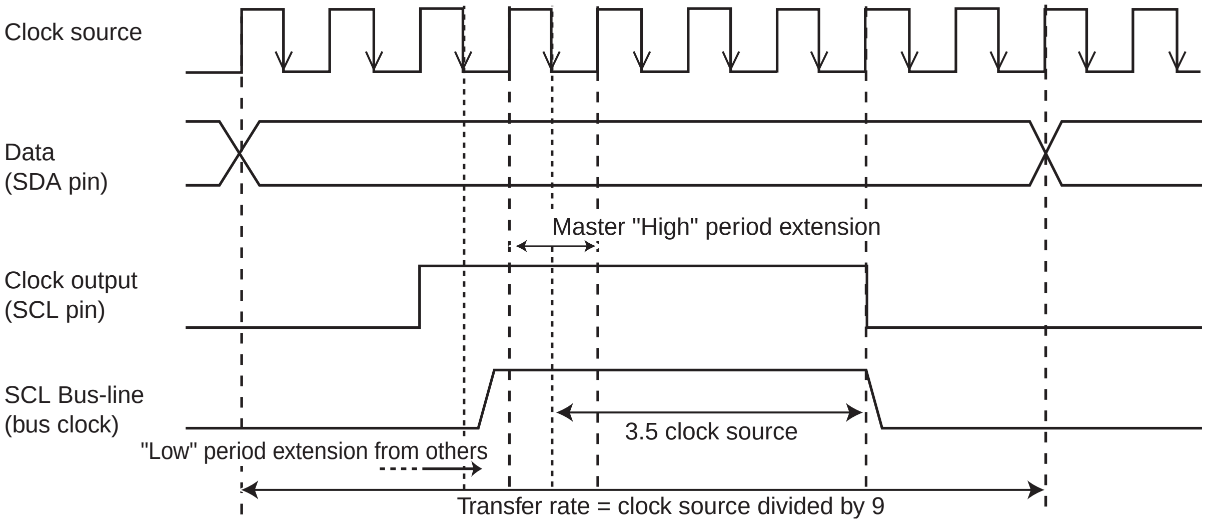

Clock source
Data
(SDA pin)
Master "High" period extension
Clock output
(SCL pin)
SCL Bus-line
(bus clock)
3.5 clock source
"Low" period extension from others
Transfer rate = clock source divided by 9

Figure:13.5.9 With "Low" Period Extension from the Other Clock (Standard Mode)

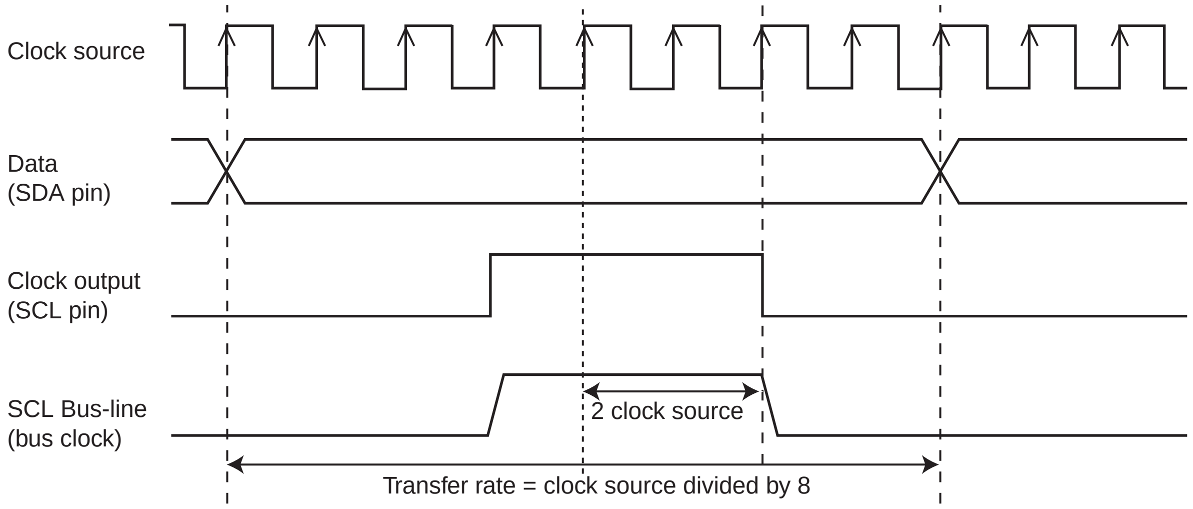


Figure:13.5.10 Without "Low" Period Extension from the Other Clock (High-speed Mode)

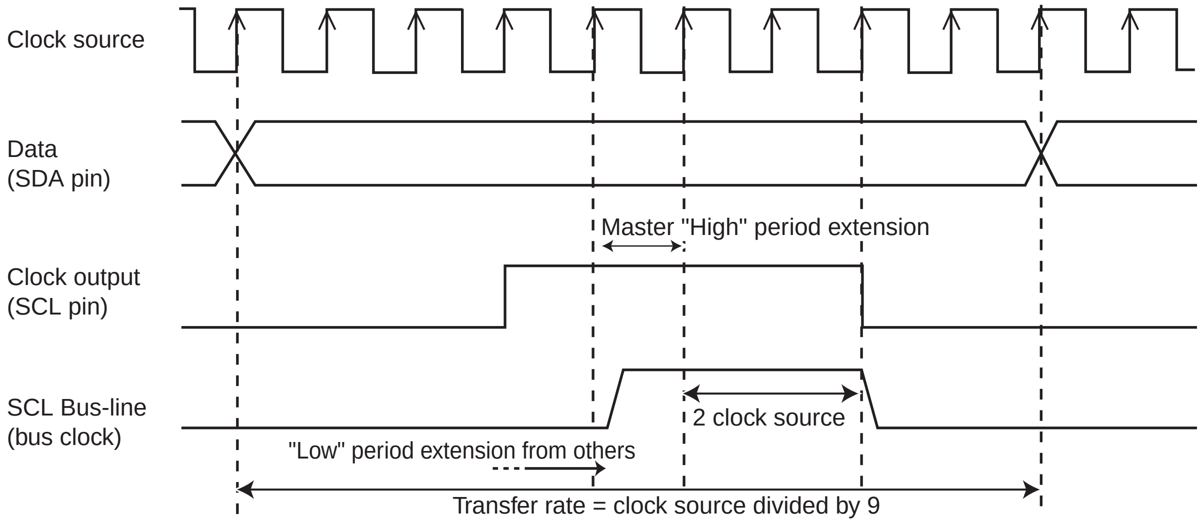

Clock source
Data
(SDA pin)
Master "High" period extension
Clock output
(SCL pin)
SCL Bus-line
(bus clock)
2 clock source
"Low" period extension from others
Transfer rate = clock source divided by 9

Figure:13.5.11 With "Low" Period Extension from the Other Clock (High-speed Mode))

To obtain the intended transfer rate, design the bus clock so that the rising time of the SCL signal does not exceed 0.5 clock (standard mode) or 1 clock (high-speed mode) of the clock source.

The following items are the same as the clock synchronous serial interface. Refer to the following pages.

■ First Transfer Bit Setup

Refer to: XIII-38

■ Transmission Data Buffer

Refer to: XIII-38

■ Reception Data Buffer

Refer to: XIII-39

■ Transmission Bit Count and First Transfer Bit

Refer to: XIII-39

■ Emergency Reset

Refer to: XIII-44

■ Master Transmission Timing

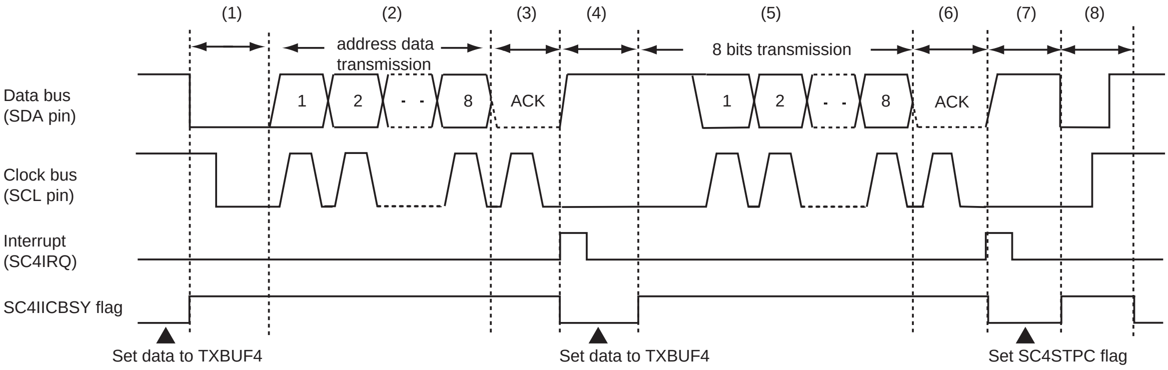


Figure:13.5.12 Master Transmission Timing

(1) Start condition output

- The SC4IICBSY flag is set to "1" after the elapse of 1 to 2 clocks of the clock source when the data is set to TXBUF4.

(2) Address data output

(3) Bus released period, ACK bit reception

(4) Interrupt process

- Communication start: set data to TXBUF4.
- The SC4IICBSY flag is set to "1" after the elapse of 1 to 10 clocks of the clock source when the data is set to TXBUF4.

(5) Transmission data output

(6) Bus released period, ACK bit reception

(7) Interrupt process

- Communication end: set the SC4STPC flag.
- The SC4IICBSY flag is set to "1" after the elapse of 1 to 2 clocks of the clock source when the SC4STPC flag is set.

(8) Stop condition generation

■ Master Reception Timing

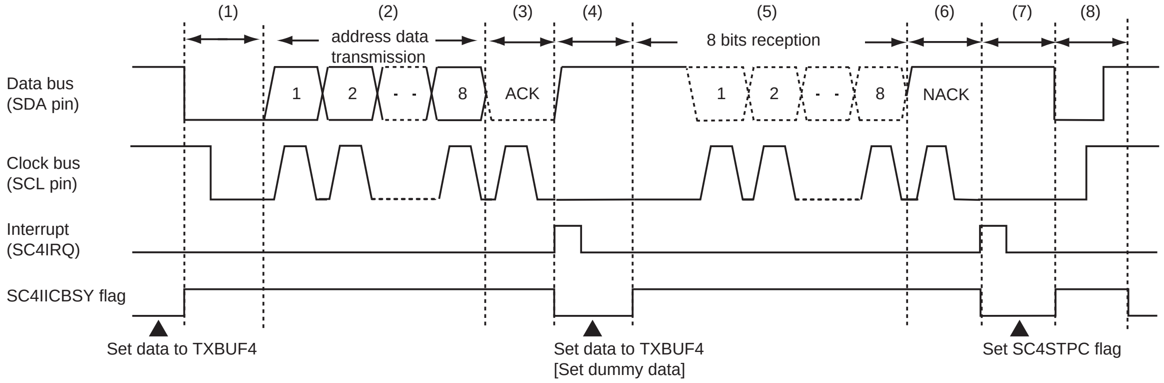


Figure:13.5.13 Master Reception Timing

(1) Start condition output

(2) Address data output

(3) Bus released period, ACK bit reception

(4) Interrupt process

- Reception mode setup: SC4REX =0→1
- Communication start: set dummy data to TXBUF4.
- The SC4IICBSY flag is set to "1" after the elapse of 1 to 10 clocks of the clock source when the dummy data is set to TXBUF4.

(5) Bus released period, data reception

(6) NACK bit output

(7) Interrupt process

- Communication end: set the SC4STPC flag.
- The SC4IICBSY flag is set to "1" after the elapse of 1 to 2 clocks of the clock source when the SC4STPC flag is set.

(8) Stop condition generation

■ Slave Transmission Timing

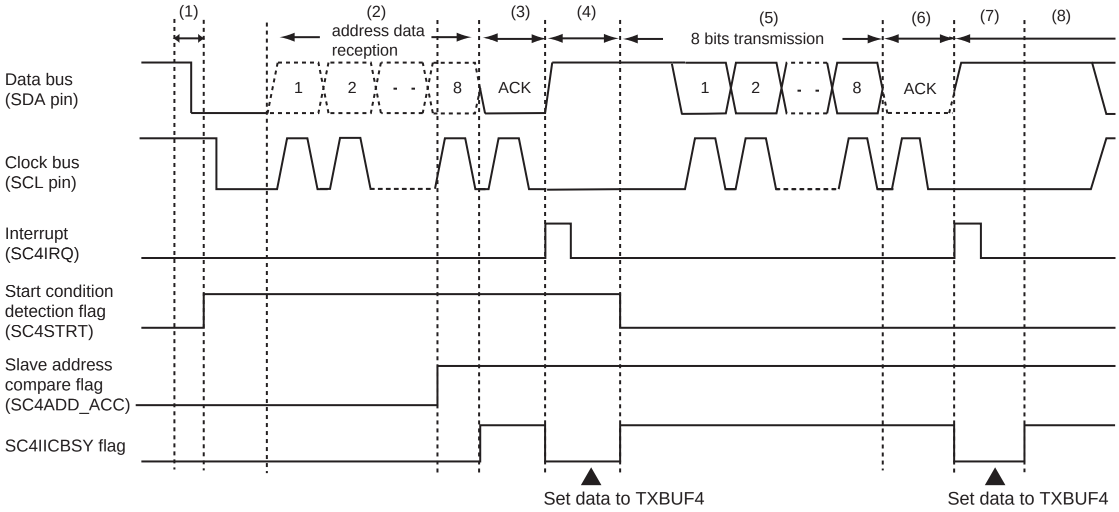


Figure:13.5.14 Slave Transmission Timing

(1) Bus released period, start condition detection

(2) Bus released period, address data reception

(3) ACK bit output

(4) Interrupt process

- Communication start: Set data to TXBUF4.
- The SC4IICBSY flag is set to "1" after the elapse of 1 to 10 clocks of the clock source when the data is set to TXBUF4.

(5) Transmission data output

(6) Bus released period, ACK bit reception

(7) Interrupt process

- Communication start: Set data to TXBUF4.
- The SC4IICBSY flag is set to "1" after the elapse of 1 to 10 clocks of the clock source when the data is set to TXBUF4.

(8) Transmission data output

■ Slave Transmission Timing (NACK Reception)

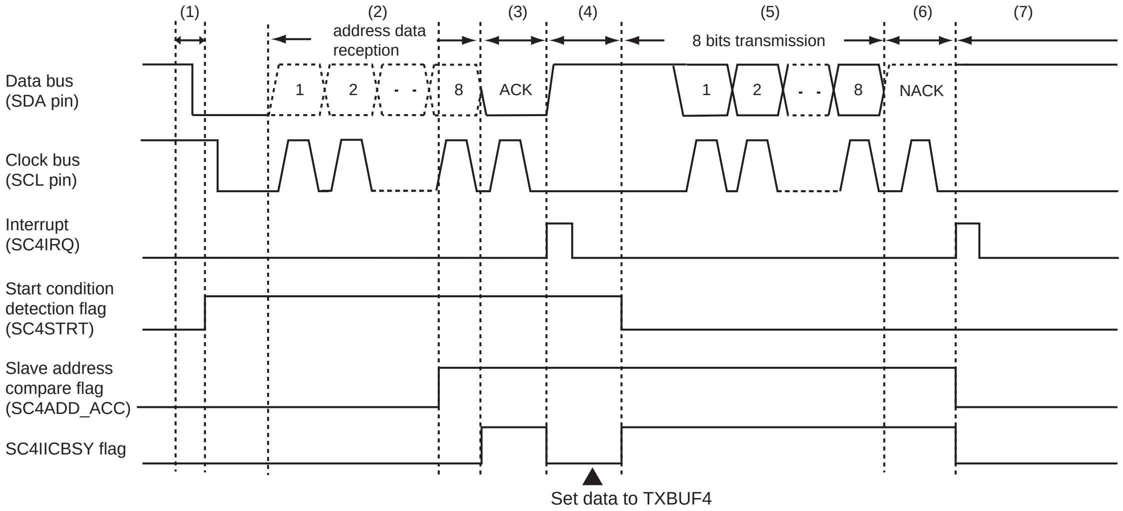


Figure:13.5.15 Slave Transmission Timing (NACK Reception)

(1) Bus released period, start condition detection

(2) Bus released period, address data reception

(3) ACK bit output

(4) Interrupt process

- Communication start: Set data to TXBUF4.
- The SC4IICBSY flag is set to "1" after the elapse of 1 to 10 clocks of the clock source when the data is set to TXBUF4.

(5) Transmission data output

(6) Bus released period, NACK bit reception

(7) Bus released period

■ Slave Reception Timing (Stop Condition Detection)

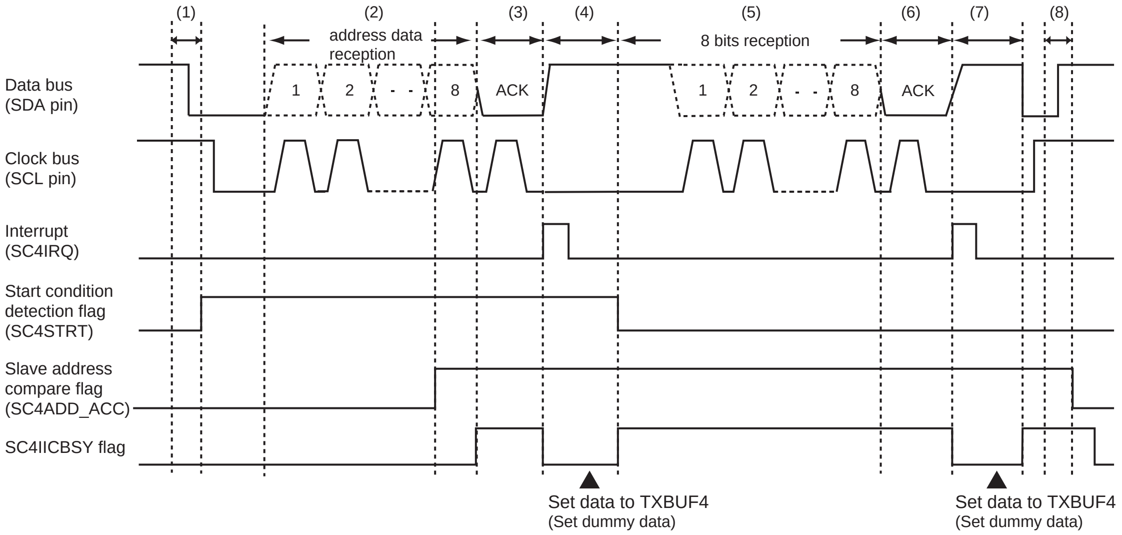


Figure:13.5.16 Slave Reception Timing (Stop Condition Detection)

(1) Bus released period, start condition detection

(2) Bus released period, address data reception

(3) ACK bit output

(4) Interrupt process

- Communication start: Set dummy data to TXBUF4.
- The SC4IICBSY flag is set to "1" after the elapse of 1 to 10 clocks of the clock source when the dummy data is set to TXBUF4.

(5) Bus released period, data reception

(6) ACK bit output

(7) Interrupt process

- Communication start: Set dummy data to TXBUF4.
- The SC4IICBSY flag is set to "1" after the elapse of 1 to 10 clocks of the clock source when the dummy data is set to TXBUF4.

(8) Stop condition detection

■ Slave Reception Timing (Restart Condition Detection)

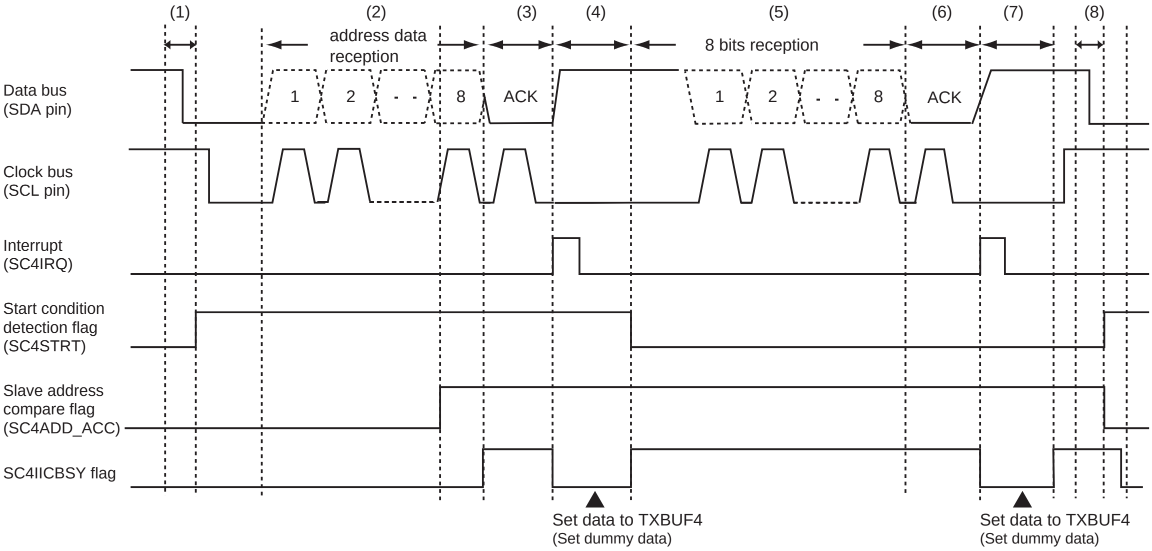


Figure:13.5.17 Slave Reception Timing (Restart Condition Detection)

(1) Bus released period, start condition detection

(2) Bus released period, address data reception

(3) ACK bit output

(4) Interrupt process

- Communication start: Set dummy data to TXBUF4.
- The SC4IICBSY flag is set to "1" after the elapse of 1 to 10 clocks of the clock source when the dummy data is set to TXBUF4.

(5) Bus released period, data reception

(6) ACK bit output

(7) Interrupt process

- Communication start: Set dummy data to TXBUF4.
- The SC4IICBSY flag is set to "1" after the elapse of 1 to 10 clocks of the clock source when the dummy data is set to TXBUF4.

(8) Restart condition detection

* Serial data input selection

13.5.2 Pin Setup

■ IIC (Multi Master) Serial Interface 4 Pin Setup

Port	Wire system	Type	Pin	Pin setup (flag setup)							
				SC0SEL register	PnDIR register	PnODC register	PnPLUD register	SC4MD1 register			
				Serial 0 I/O pin switching	I/O mode selection	Nch open-drain output selection Arbitrary setting	Pull-up /Pull-down resistor selection Aribitary setting	Serial data input selection	SBT0 pin function selection	Serial input control selection	SBO4 (SDA4) pin function selection
				0:A type 1:B type	0: input mode 1: output mode	0: push/ pull 1: Nch open-drain	0: not added 1: added	0: data input from SBI4 1: data input from SBO4	0: port 1: transfer clock I/O	0:"1" input 1: serial input	0: port 1: serial data output
				0SL0	PnDIRm	PnODCm	PnPLUm	SC0IOM	SC0SBTS	SC0SBIS	SC0SBOS
Port 7	2-wire	Transmission	P71/SDA4A	0	P7DIR1:1	P7ODC1	P7PLU1	1	1	1	1
			P72/SCL4A		P7DIR2:1	P7ODC2	P7PLU2				
		Reception	P71/SDA4A		P7DIR1:0	-	-	1	1	1	1
			P72/SCL4A		P7DIR2:1	P7ODC2	P7PLU2				
Port 3	2-wire	Transmission	P33/SDA4B	1	P3DIR3:1	P3ODC3	P3PLUD3	1	1	1	1
			P34/SCL4B		P3DIR4:1	P3ODC4	P3PLUD4				
		Reception	P33/SDA4B		P3DIR3:0	-	-	1	1	1	1
			P34/SCL4B		P3DIR4:1	P3ODC4	P3PLUD4				

13.5.3 Setup Example

■ Master Transmission Setup Example

The following describes the setup example for multiple data transmission to all the devices on IIC bus using Serial Interface 4 multi master IIC interface function. Table:13.5.6 shows communication conditions.

Table:13.5.6 Setup Conditions of Multi Master IIC Communication

Setup item	Set to
Serial data input selection	SDA4
Transfer bit count	8 bits
Start condition	Enabled (after 2nd communication: disabled)
First transfer bit	MSB
ACK bit	Enabled
IIC communication mode	Standard mode
Clock source	fpll-div/32
SCL4/SDA4 pin type	Nch open-drain
SCL4 pin pull-up resistor	Added
SDA4 pin pull-up resistor	Added
Master/Slave setting	Master

An example setup procedure is shown below.

Setup Procedure	Description
(1) Select the prescaler operation SC4MD2(0x03F52) bp3: SC4PSCE =1	(1) Set the SC4PSCE flag of SC4MD2 register to "1" to select "Enable count" for prescaler count control.
(2) Select the clock source SC4MD2(0x03F52) bp2 to 0: SC4PSC2 to 0 =011	(2) Select the clock source by SC4MD2 register. Set SC4PSC2 to 0 flags to "011" to select fpll-div/32.
(3) Control the pin type P7ODC(0x03EF7) bp1: P7ODC1 =1 bp2: P7ODC2 =1 P7PLU(0x03EA7) bp1: P7PLU1 =1 bp2: P7PLU2 =1	(3) Set the P7ODC1 and P7ODC2 flags of P7ODC register to "11" to select Nch open-drain for SDA4/SCL4. Set the P7PLU1 and P7PLU2 flags of P7PLU register to "11" to add pull-up resistor.
(4) Control the pin direction. P7DIR(0x03E97) bp1: P7DIR1 =1 bp2: P7DIR2 =1	(4) Set the P7DIR1 and P7DIR2 flags of P7DIR register to "11" to set P71 and P72 to output mode.

Setup Procedure	Description
(5) Set SC4MD3 register Set ACK bit SC4MD3(0x03F53) bp0: SC4ACK0 =X bp1: SC4ACKS =1 Set the communication mode. SC4MD3(0x03F53) bp4: SC4TMD =0 Select the communication type. SC4MD3(0x03F53) bp2: SC4CMD =1 Select Transmission/Reception Select transmission/reception mode. SC4MD3(0x03F53) bp3: SC4REX =0	(5) Set the SC4ACKS flag of Serial Interface 4 mode register 3 (SC4MD3) to "1" to select “ACK bit enabled”. ACK bit is received at transmission and the level setting of ACK bit with the SC4ACKO flag is not needed.Set the SC4TMD flag of SC4MD3 register to "0" to set standard mode.Set the SC4CMD flag of SC4MD3 register to "1" to select IIC. Set the SC4REX flag of SC4MD3 register to "0" to select transmission mode. The setting to Serial Interface 4 mode register 3 (SC4MD3) should be executed with one-time MOV instruction in accordance with the setting of each bit. Do not set with BSET/BCLR. Also, do not read out the value of SC4MD3 and write to the specified bit. SC4ACKO flag will be destroyed.
(6) Initialize the monitor flag SC4STR1(0x03F57) bp0: SC4DATA_ERR =0	(6) Set the SC4DATA_ERR flag of Serial Interface 4 status register 1 (SC4STR1) to "0" to initialize the communication error detection flag.
(7) Set SC4MD0 register Select the transfer bit count SC4MD0(0x03F50) bp2 to 0: SC4LNG2 to 0 =111 Select the start condition SC4MD0(0x03F50) bp3: SC4STE =0 Select the first bit to be transferred SC4MD0(0x03F50) bp4: SC4DIR =0	(7) Set the SC4LNG2 to 0 flags of Serial Interface 4 mode register 0 (SC4MD0) to "111" to set the transfer bit count to 8 bits. Set the SC4STE flag of SC4MD0 register to "0" to disable the stand condition (start condition is not added after second communication). Set the SC4DIR flag of SC4MD0 register to "0" to set the first transfer bit to MSB.
(8) Set SC4MD1 register Select the transfer clock SC4MD1(0x03F51) bp2: SC4MST =1 Control the pin function SC4MD1(0x03F51) bp4: SC4SBOS =1 bp5: SC4SBIS =1 bp6: SC4SBTS =1 bp7: SC4IOM =1	(8) Set the SC4MST flag of SC4MD1 register to "1" to select the clock master (internal clock). Set the SC4SBOS, SC4SBIS and SC4SBTS flags of SC4MD1 register to "1" to set SDA4 (SBO4) pin to serial data output, SBI4 pin to serial data input, and SCL4 (SBT4) pin to serial clock I/O. Set the SC4IOM flag to "1" to set SDA4 (SBO4) pin to serial data input.
(9) Set the interrupt level PSW bp6: MIE =0 PERIILR(0x03FFE) bp7 to 6: PERILV1 to 0 =10	(9) Set the MIE flag of PSW to "1" to enable maskable interrupts. Set the interrupt level by the PERILV1 to 0 flags of Peripheral function group interrupt control register PERIILR.

Setup Procedure	Description
(10) Enable the interrupt IRQEXPDT(0x03F4F) bp5 to 4: IRQEXPDT5 to 4 = 00 IRQEXPEN(0x03F4E) bp5 to 4: IRQEXPEN5 to 4 =11 PSW bp6: MIE =1	(10) Read the IRQEXPDT5 to 4 flags of the peripheral function group interrupt factor retention register (IRQEXPDT) and write the data to clear the peripheral function group request flag. To execute interrupt processing before this setting, do not clear the corresponding flag of the IRQEXPDT register. Set the IRQEXPEN5 to 4 flags of the peripheral function group interrupt input enable register (IRQEXPEN) to enable the corresponding interrupt. [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup] Set the MIE flag of PSW to “1” to enable maskable interrupts.
(11) <Transmission starts> Start the serial transmission. Confirm that SCL4 is "High" Transmission data→TXBUF4(0x03F59)	(11) Set the transmission data in the transmission buffer TXBUF4. A transfer clock is generated and transmission starts. After data is transmitted, if ACK bit is received, Serial 4 communication completion interrupt SC4IRQ is generated.
(12) <Transmission ends> <Setup for the next data transmission> Determine the monitor flag SC4STR1(0x03F57) bp0: SC4DATA_ERR	(12) Check the SC4DATA_ERR flag of Serial Interface 4 status register 1 (SC4STR1). When the previous transmission is normally completed, SC4DATA_ERR = "0". If SC4DATA_ERR = "1", reexecute the communication.
(13) Determine the ACK bit level SC4MD3(0x03F53) bp0: SC4ACK0	(13) Check the ACK bit level received by the SC4ACK0 flag of Serial Interface 4 mode register 3 (SC4MD3). When SC4ACK0=0, transmission continues. When SC4ACK0=1, the slave side may not receive data. In this case, finish the communication.
(14) Set SC4MD0 register Select the transfer bit count SC4MD0(0x03F50) bp2 to 0: SC4LNG2 to 0	(14) When changing the transfer bit count, set the transfer bit count by the SC4LNG2 to 0 flags of Serial Interface 4 mode register 0 (SC4MD0).
(15) <The next data transmission starts> Serial transmission starts [→(14)]	(15) Set the transmission data in TXBUF4 to start transmission.[→(14)]
(16) <Transmission ends> <End process of IIC communication> Set the SC4STPC flag SC4MD3(0x03F53) bp5: SC4STPC =1	(16) Set the SC4STPC flag of Serial Interface 4 mode register 3 (SC4MD3) to "1". A stop condition is generated automatically and communication ends.

* (1) and (2) can be set at once.
* Each setup in (7) and (8) can be set at once.
* (9) to (10) can be set at once.

Set each flag in accordance with the order of the setup procedure. Activate the communication after all the control registers (Table:13.2.1, except TXBUF4) are set.

■ Slave Transmission Setup

The following describes the setup examples for slave transmission processing using serial 4 IIC interface function. Table:13.5.7 shows the conditions for transmission processing.

Table:13.5.7 Setup Conditions of Slave IIC Communication

Item	Set to
Serial data input selection	SDA4
Transfer bit count	8 bits
First transfer bit	MSB
ACK bit	Enabled
IIC communication mode	Standard mode
Clock source	fpll-div/32
SCL4/SDA4 pin type	Nch open-drain
SCL4 pin pull-up resistor	Added
SDA4 pin pull-up resistor	Added
Address mode	7 bits
Slave address	0110011
Master/slave setup	Master (multi-master)

An example setup procedure is shown below.

Setup Procedure	Description
(1) Select the prescaler operation SC4MD2(0x03F52) bp3: SC4PSCE =1	(1) Set the SC4PSCE flag of SC4MD2 register to "1" to select "Enable count" for prescaler count control.
(2) Select the clock source SC4MD2(0x03F52) bp2 to 0: SC4PSC2 to 0 =011	(2) Select clock source by SC4MD2 register. Set bp2 to 0 to "011" to select fpll-div/32.
(3) Control the pin type P7ODC(0x03EF7) bp1: P7ODC1 = 1 bp2: P7ODC2 = 1 P7PLU(0x03EA7) bp1: P7PLU1 = 1 bp2: P7PLU2 = 1	(3) Set the P7ODC1 and P7ODC2 flags of P7ODC register to "11" to select Nch open-drain for SDA4/SCL4. Set the P7PLU1 and P7PLU2 flags of P7PLU register to "11" to add pull-up resistor.
(4) Control the pin direction. P7DIR(0x03E97) bp1: P7DIR1 =1 bp2: P7DIR2 =1	(4) Set the P7DIR1 and P7DIR2 flags of P7DIR register to "11" to set P71 and P72 to output mode.

Setup Procedure	Description
(5) Set the ACK bit SC4MD3(0x03F53) bp0: SC4ACK0 =X bp1: SC4ACKS =1 Set the communication mode SC4MD3(0x03F53) bp4: SC4TMD =0 Select the communication type SC4MD3(0x03F53) bp2: SC4CMD =1	(5) Set the SC4ACKS flag of Serial Interface 4 mode register 3 (SC4MD3) to "1" to select “ACK bit enabled”. ACK bit is received at transmission and the level setting of ACK bit with the SC4ACKO flag is not needed. Set the SC4TMD flag of SC4MD3 register to "0" to set standard mode. The setting to Serial Interface 4 mode register 3 (SC4MD3) should be executed with one-time MOV instruction in accordance with the setting of each bit. Do not set with BSET/BCLR. Also, do not read out the value of SC4MD3 and write to the specified bit. SC4ACKO flag will be destroyed.
(6) Initialize the monitor flag SC4STR1(0x03F57) bp0: SC4DATA_ERR =0	(6) Set the SC4DATA_ERR flag of Serial Interface 4 status register 1 (SC4STR1) to "0" to initialize the communication error detection flag.
(7) Set SC4MD0 register Select transfer bit count SC4MD0(0x03F50) bp2 to 0: SC4LNG2 to 0 =111 Select the first transfer bit SC4MD0(0x03F50) bp4: SC4DIR =0	(7) Set the SC4LNG2 to 0 flags of Serial Interface 4 mode register 0 (SC4MD0) to "111" to set the transfer bit count to 8 bits. Set the SC4DIR flag of SC4MD0 register to "0" to set the first transfer bit to MSB.
(8) Set SC4MD1 register Select the transfer clock SC4MD1(0x03F51) bp2: SC4MST =1 Control the pin function SC4MD1(0x03F51) bp4: SC4SBOS =1 bp5: SC4SBIS =1 bp6: SC4SBTS =1 bp7: SC4IOM =1	(8) Set the SC4MST flag of SC4MD1 register to "1" to select the clock master (internal clock). Set the SC4SBOS, SC4SBIS, and SC4SBTS flags of SC4MD1 register to "1" to set SDA4 (SBO4) pin to serial data output, SBI4 pin to serial data input, and SCL4 (SBT4) pin to serial clock I/O. Set the SC4IOM flag to "1" to set SDA4 (SBO4) pin to serial data input.
(9) Set the slave address SC4AD0(0x03F54) bp7 to 1: SC4ADR7 to 1 = 0110011	(9) Set the slave address tin the upper 7 bits (SC4ADR7 to 1) of SC4AD0 register.
(10) Set the interrupt level PSW bp6: MIE =0 PERIILR(0x03FFE) bp7 to 6: PERIV1 to 0 =10	(10) Set the MIE flag of PSW to "1" to enable maskable interrupts. Set the interrupt level by the PERILV1 to 0 flags of Peripheral function group interrupt control register (PERIICR).
(11) Enable the interrupt IRQEXPDT(0x03F4F) bp5 to 4: IRQEXPDT5 to 4 = 00 IRQEXPEN(0x03F4E) bp5 to 4: IRQEXPEN5 to 4 = 11 PSW bp6: MIE =1	(11) Read the IRQEXPDT5 to 4 flags of the peripheral function group interrupt factor retention register (IRQEXPDT) and write the data to clear the peripheral function group request flag. To execute interrupt processing before this setting. do not clear the corresponding flag of the IRQEXPDT register. Set the IRQEXPEN5 to 4 flags of the peripheral function group interrupt input enable register (IRQEXPEN) to enable the corresponding interrupt. [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup] Set the MIE flag of PSW to “1” to enable maskable interrupts.

Setup Procedure	Description
(12) Start IIC communication.	(12) The master on the IIC bus starts communication.
(13) Check the data transmission/reception SC4STR1(0x03F57) bp7: SC4WRS = 1	(13) Serial 4 communication completion interrupt (SC4IRQ) is generated when the address transmitted from the master matches the slave address set in SC4AD0 register. In the interrupt handling routine, the communication is recognized as the slave transmission by verifying that the SC4WRS flag of SC4STR1 register is set to "1".
(14) Set the transmission data TXBUF4(0x03F59) bp7 to 0: TXBUF47 to 0 = 0x55	(14) Set the data in TXBUF4 register.

* (1) and (2) can be set at once.
* Each setup in (7) and (8) can be set at once.
* (9) to (10) can be set at once.

Set each flag in accordance with the order of the setup procedure. Activate the communication after all the control registers (Table:13.2.1, except TXBUF4) are set.

Chapter 14 A/D Converter

14.1 Overview

This LSI has an A/D converter with 10 bits resolutions. It contains a built-in sample hold circuit.
The channels 0 to 15 (AN0 to AN15) of analog input can be switched by software.
When A/D converter is stopped, the power consumption can be reduced by turning the built-in ladder resistance OFF. A/D converter is activated by a register set and an external interrupt.
Moreover, the touch sensor of an electric capacity system is realizable using an A/D converter.

14.1.1 Functions

Table:14.1.1 shows the A/D converter functions.

Table:14.1.1 A/D Converter Functions

A/D Input Pins	16 pins (12 pins) *1
Pins	AN15 to AN0 (AN11 to AN0) *1
Interrupt	ADIRQ
Resolution	10 bits
Conversion Time (Minimum)	12.93 μs (T_{AD}= as 800 ns)
Input Range	V_{SS} to V_{REF+}
Power Consumption	Built-in Ladder Resistance (ON/OFF)

*1 ( ) denotes pin specification of MN101EFA7/A2.

This function can not be used in STOP/HALT mode.

To realize a low power consumption, we recommend that the built-in ladder resistance is turned OFF while A/D conversion is not executed.

14.1.2 Block Diagram

■ A/D Converter Block Diagram

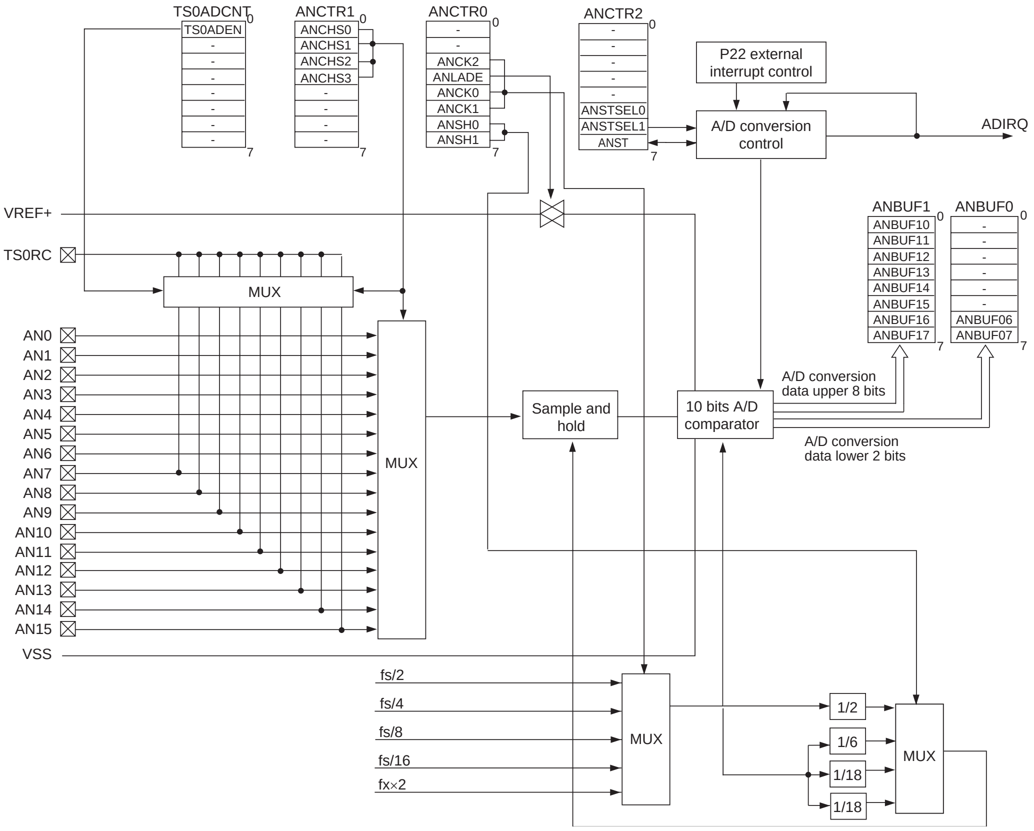


Figure:14.1.1 A/D Converter Block Diagram (MN101EFA8)

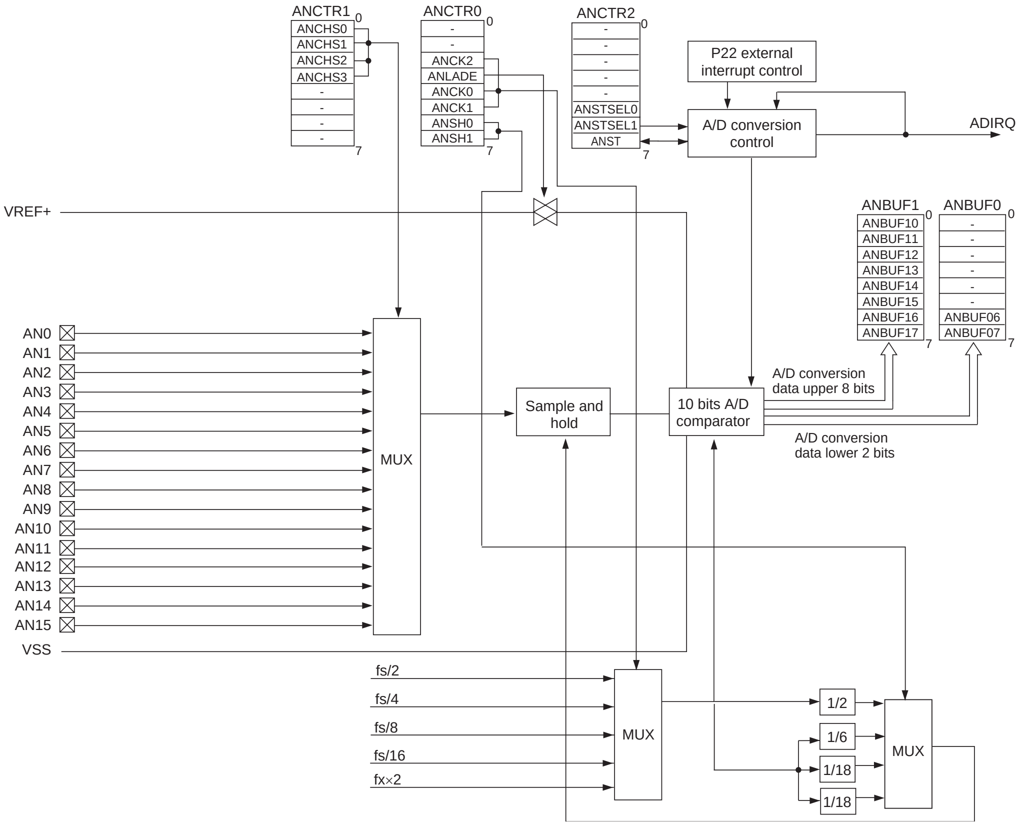


Figure:14.1.2 A/D Converter Block Diagram (MN101EFA3)

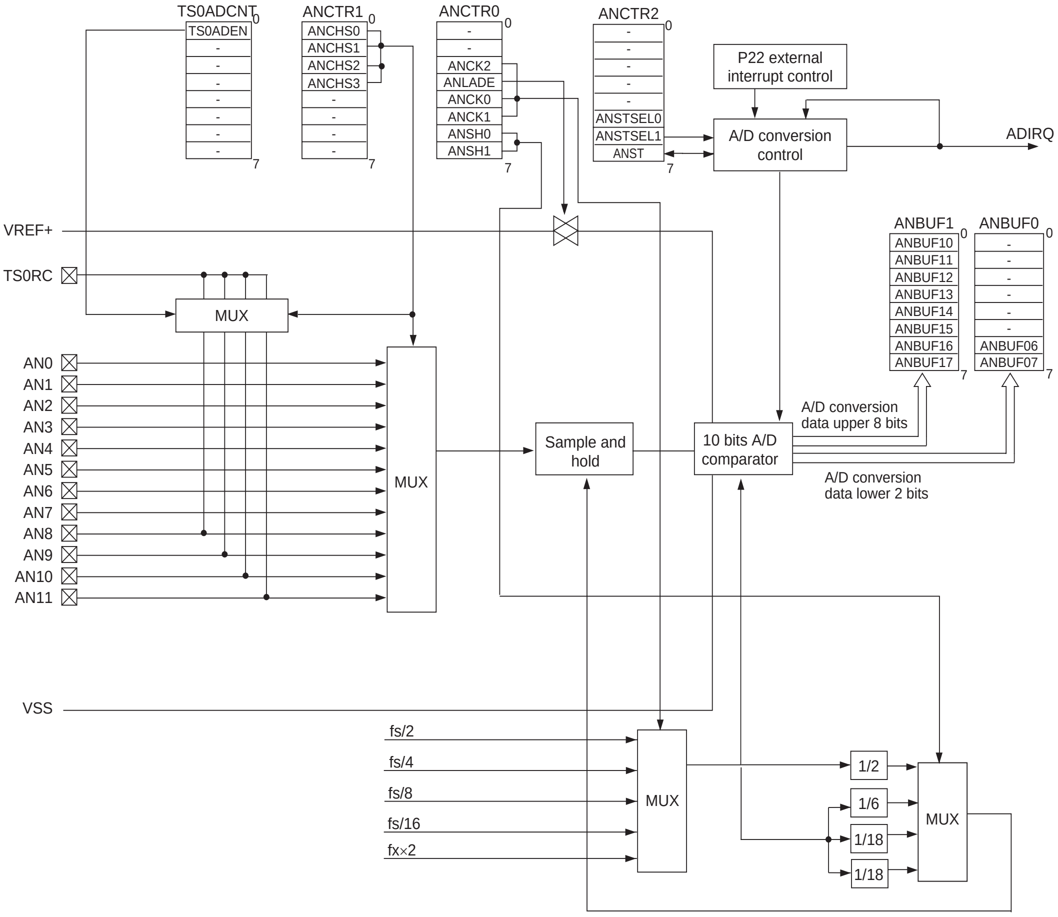


Figure:14.1.3 A/D Converter Block Diagram (MN101EFA7)

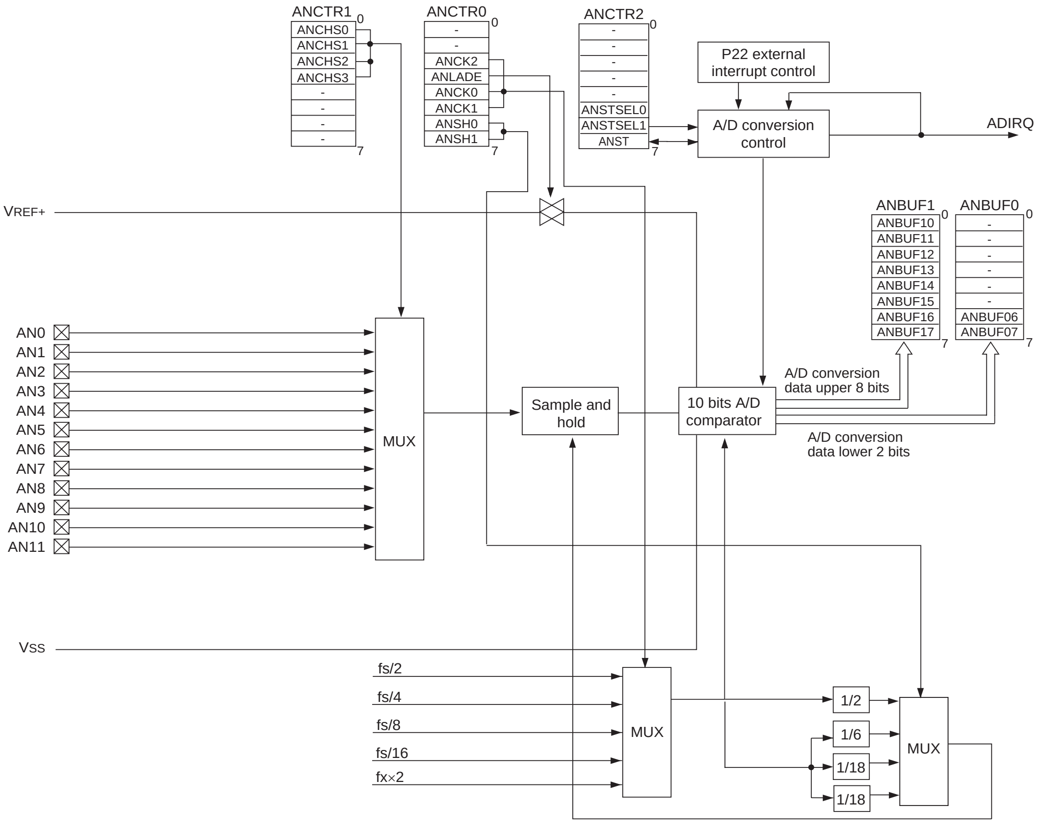


Figure:14.1.4 A/D Converter Block Diagram (MN101EFA2)

14.2 Control Registers

A/D converter consists of the control register (ANCTRn) and the data storage buffer (ANBUFn).

14.2.1 Registers

Table:14.2.1 shows the registers used to control A/D converter.

Table:14.2.1 A/D Converter Control Registers

MN101EFA8/A3 Table remarks √: With function -: Without function

Register	Address	R/W	Function	Page	MN101EFA8	MN101EFA3
ANCTR0	0x03FC5	R/W	A/D converter control register 0	XIV-9	√	√
ANCTR1	0x03FC6	R/W	A/D converter control register 1	XIV-10	√	√
ANCTR2	0x03FC7	R/W	A/D converter control register 2	XIV-11	√	√
ANBUF0	0x03FC8	R	A/D converter data storage buffer 0	XIV-12	√	√
ANBUF1	0x03FC9	R	A/D converter data storage buffer 1	XIV-12	√	√
ADICR	0x03FFD	R/W	A/D conversion interrupt control register	XIV-9	√	√
EDGDT	0x03F1E	R/W	Both edges interrupt control register	IV-46	√	√
P3IMD	0x03EC3	R/W	Port 3 input mode register	V-28	√	√
P3PLUD	0x03EA3	R/W	Port 3 pull-up/pull-down resistor control register	V-27	√	√
P9IMD	0x03EC9	R/W	Port 9 input mode register	V-96	√	√
P9PLUD	0x03EA9	R/W	Port 9 pull-up/pull-down resistor control register	V-95	√	√
PAIMD	0x03ECA	R/W	Port A input mode register	V-110	√	√
PAPLU	0x03EAA	R/W	Port A pull-up resistor control register	V-109	√	√
PBIMD	0x03ECB	R/W	Port B input mode register	V-118	√	-
PBPLUD	0x03EAB	R/W	Port B pull-up/pull-down resistor control register	V-118	√	-
TS0ADCNT	0x03DA5	R/W	Touch sensor 0 A/D control enable register	XIV-13	√	-

R/W: Readable/Writable
R: Read only

MN101EFA7/A2 Table remarks √: With function -: Without function

Register	Address	R/W	Function	Page	MN101EFA7	MN101EFA2
ANCTR0	0x03FC5	R/W	A/D converter control register 0	XIV-9	√	√
ANCTR1	0x03FC6	R/W	A/D converter control register 1	XIV-10	√	√
ANCTR2	0x03FC7	R/W	A/D converter control register 2	XIV-11	√	√
ANBUF0	0x03FC8	R	A/D converter data storage buffer 0	XIV-12	√	√
ANBUF1	0x03FC9	R	A/D converter data storage buffer 1	XIV-12	√	√
ADICR	0x03FFD	R/W	A/D conversion interrupt control register	IV-30	√	√
EDGDT	0x03F1E	R/W	Both edges interrupt control register	IV-46	√	√
P5IMD	0x03EC5	R/W	Port 5 input mode register	V-48	√	√
P5PLUD	0x03EA5	R/W	Port 5 pull-up/pull-down resistor control register	V-47	√	√
P9IMD	0x03EC9	R/W	Port 9 input mode register	V-96	√	√
P9PLUD	0x03EA9	R/W	Port 9 pull-up/pull-down resistor control register	V-95	√	√
PAIMD	0x03ECA	R/W	Port A input mode register	V-110	√	√
PAPLU	0x03EAA	R/W	Port A pull-up resistor control register	V-109	√	√
TS0ADCNT	0x03DA5	R/W	Touch sensor 0 A/D control enable register	XIV-13	√	-

R/W: Readable/Writable
R: Read only

14.2.2 Control Registers

■ A/D Converter Control Register 0 (ANCTR0: 0x03FC5)

bp	7	6	5	4	3	2	1	0
Flag	ANSH1	ANSH0	ANCK1	ANCK0	ANLADE	ANCK2	-	-
Reset	0	0	0	0	0	0	-	-
Access	R/W	R/W	R/W	R/W	R/W	R/W	-	-

bp	Flag	Description
7-6	ANSH1-0	Sample and hold time 00: $T_{AD} \times 2$ 01: $T_{AD} \times 6$ 10: $T_{AD} \times 18$ 11: $T_{AD} \times 18$
5-4	ANCK2-0	A/D conversion clock (ftad=1/T_{AD}) 000: fs/2 001: fs/4 010: fs/8 011: fx × 2 100: fs/16 101: fs/16 110: fs/16 111:Setting prohibited * as 800 ns $\leq T_{AD} \leq$ 15.26 µs
3	ANLADE	A/D ladder resistance control 0: A/D ladder resistance OFF 1: A/D ladder resistance ON
1-0	-	-

■ A/D Converter Control Register 1 (ANCTR1: 0x03FC6)

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	ANCHS3	ANCHS2	ANCHS1	ANCHS0
Reset	-	-	-	-	0	0	0	0
Access	-	-	-	-	R/W	R/W	R/W	R/W

bp	Flag	Description
7-4	-	-
3-0	ANCHS3-0	Analog input channel 0000: AN0 0001: AN1 0010: AN2 0011: AN3 0100: AN4 0101: AN5 0110: AN6 0111: AN7 1000: AN8 1001: AN9 1010: AN10 1011: AN11 1100: AN12 (Setting prohibited) * 1101: AN13 (Setting prohibited) * 1110: AN14 (Setting prohibited) * 1111: AN15 (Setting prohibited) *

* ( ) denotes pin specification of MN101EFA7/A2.

■ A/D Converter Control Register 2 (ANCTR2: 0x03FC7)

bp	7	6	5	4	3	2	1	0
Flag	ANST	ANST SEL1	ANST SEL0	-	-	-	-	-
Reset	0	0	0	-	-	-	-	-
Access	R/W	R/W	R/W	-	-	-	-	-

bp	Flag	Description
7	ANST	A/D conversion status 0: Finish, Stop 1: Start, Converting
6-5	ANSTSEL1-0	A/D conversion starting factor selection 00: Set ANST flag to "1" 01: Set ANST flag to "1" 10: Set P22 external interrupt, or ANST flag to "1" 11: Set A/D conversion interrupt, or ANST flag to "1"
4-0	-	-

14.2.3 Data Buffers

■ A/D Converter Data Storage Buffer 0 (ANBUF0: 0x03FC8)

The lower 2 bits results from A/D conversion are stored to this register.

bp	7	6	5	4	3	2	1	0
Flag	ANBUF07	ANBUF06	-	-	-	-	-	-
Reset	X	X	-	-	-	-	-	-
Access	R	R	-	-	-	-	-	-

■ A/D Converter Data Storage Buffer 1 (ANBUF1: 0x03FC9)

The upper 8 bits results from A/D conversion are stored to this register.

bp	7	6	5	4	3	2	1	0
Flag	ANBUF17	ANBUF16	ANBUF15	ANBUF14	ANBUF13	ANBUF12	ANBUF11	ANBUF10
Reset	X	X	X	X	X	X	X	X
Access	R	R	R	R	R	R	R	R

Do not execute the word access to ANBUF1 register.

14.2.4 Touch Sensor 0 A/D Control Enable Register

■ Touch Sensor 0 A/D Control Enable Register (TS0ADCNT: 0x03DA5)

Touch Sensor 0 A/D control enable register connects pin TS0RC to the analog input channel selected by ANCTR1 register. Set up this register to operate touch sensor using A/D converter,

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	-	-	-	TS0ADEN
Reset								0
Access								R/W

bp	Flag	Description
7-1	-	-
0	TS0ADEN	Touch Sensor 0 A/D control enable 0: Disabled 1: Enabled

! Do not operate Touch Sensor 0 Timer while the TS0ADEN flag is “1”.

14.3 Operation

Here is a description of A/D converter circuit setup procedure.

1. Set the analog pins.

   Set the analog input pin, set in (2), to "special function pin" by the port n, input mode register (P3IMD, P5IMD, P9IMD, PAIMD, PBIMD).

   * Setup of the port n input mode register should be done before analog voltage is applied to pins.

2. Select the analog input pin.

   Select the analog input pin from AN15 to AN0 by the ANCHS3 to 0 flags of the A/D converter control register1 (ANCTR1).

3. Select the A/D conversion clock.

   Select the A/D conversion clock by the ANCK2, ANCK1, ANCK0 flags of the A/D converter control register 0 (ANCTR0). Setup should be such a way that converter clock (T_{AD}) does not drop less than 800 ns with any resonator.

4. Set the sample hold time.

   Set the sample hold time by the ANSH1, ANSH0 flag of the A/D converter control register 0 (ANCTR0). The sample hold time should be based on analog input impedance.

   * (2) to (4) are not in order. (3) and (4) can be operated simultaneously.

5. Set the A/D ladder resistance.

   Set the ANLADE flag of the A/D converter control register 0 (ANCTR0) to "1", and a current flow through the ladder resistance and A/D converter goes into the waiting.

6. Select the A/D converter activation factor, then start A/D conversion.

   Set the ANST flag of the A/D converter control register 2 (ANCTR2) to "1" to start A/D converter or set ANSTSEL1, ANSTSEL0 flags of A/D converter control register 2 (ANCTR2) to "10" to start A/D converter by the external trigger factor.

7. A/D conversion.

   A/D conversion is compared and determined sequentially by MSB after the sampling in the sample hold time to be set (4).

8. Complete the A/D conversion.

   After the A/D conversion is completed, the result of the conversion is stored in the A/D conversion data store buffer (ANBUF0, 1), the A/D conversion interrupt is generated and the ANST flag is cleared to "0".

Set the ANLADE flag to "1" then start A/D conversion after waiting for 12 conversion clock.

When A/D converter is started again after setting the ANST flag to "0" and A/D converter was stopped by force during A/D converter, start A/D converter after waiting for more than (2 system clock) + (2 converter clock) considerable time.

In the A/D conversion starting factor selection, when A/D is converted in the status that the start by the external interrupt is selected and the ANST flag is set to "0" to be completed A/D conversion forcefully during the A/D conversion, be sure to set the A/D conversion starting factor selection to "0" in advance before setting the ANST flag to "0".

If the flags of ANCTR0, ANCTR1 are changed during A/D conversion, we can not guarantee the operation and the result of A/D conversion.
Set the ANLADE flag of the A/D converter control register 0 (ANCTR0) to "0" to disable the operation to change the flags.

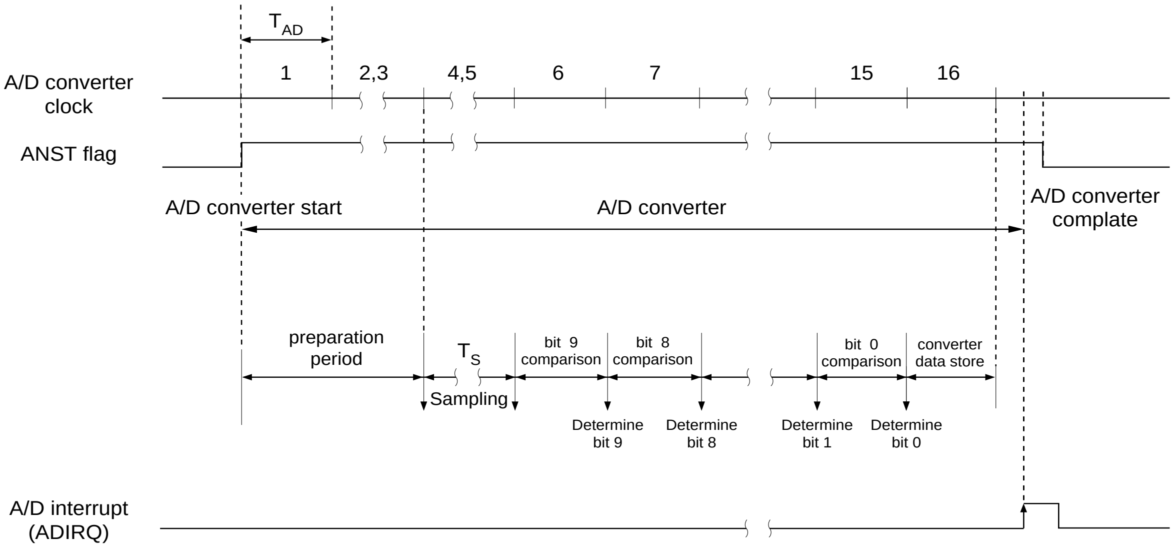


Figure:14.3.1 Operation of A/D conversion (sample hold time at T_{AD} × 2)

To read out the value of the A/D conversion, A/D conversion should be done several times to prevent noise error by confirming the match of level by program, or by using the average value.

14.3.1 Setup

■ Input Pins of A/D Conversion Setup

Input pins for A/D converter is selected by the ANCHS3 to 0 flags of the ANCTR1 register.

■ A/D Conversion Clock Setup

The A/D conversion clock is set with the ANCK2 to 0 flags of the ANCTR0 register. Set the A/D conversion cycle (T_{AD}) less than 800 ns or 15.26 µs under. Table:14.3.1 shows the machine clock (fpll, fx, fs) and the A/D conversion cycle (T_{AD}). (calculated as fs = fpll/2, fx/4)

Table:14.3.1 A/D Conversion Clock and A/D Conversion Cycle

ANCK2	ANCK1	ANCK0	A/D conversion clock	A/D conversion cycle (TAD)		
				High-speed operation		Low-speed operation
				fpll=10 MHz	fpll=8.38 MHz	fx=32.768 kHz
0	0	0	fs/2	400 ns (Setting prohibited)	477.33 ns (Setting prohibited)	244.14 µs
		1	fs/4	800 ns	954.65 ns	488.28 µs
	1	0	fs/8	1.6 µs	1.91 s	976.56 µs
		1	fx×2	15.26 µs	15.26 µs	15.26 µs
1	X	X	fs/16	3.2 µs	3.82 µs	1953.12 µs

■ A/D Conversion Sample hold Time (T_S) Setup

The sample hold time of A/D conversion is set with the ANSH1 to 0 flags of the ANCTR0 register. The sample hold time of A/D conversion depends on external circuit, so set the right value by analog input impedance.

Table:14.3.2 Sample Hold Time of A/D Conversion and A/D Conversion Time

ANSH1	ANSH0	Sample hold time (TS)	A/D conversion time [µs]				
			High-speed operation				Low-speed operation
			TAD=1.6 µs (fs=5 MHz)	TAD=954.65 ns (fs=4.19 MHz)	TAD=1.91 µs (fs=4.19 MHz)	TAD=15.26 µs (fs=4.19 MHz)	TAD=15.26 µs (fs=8.192 kHz)
0	0	TAD×2	26.1	15.87	31.16	244.88	610.37
	1	TAD×6	32.5	19.69	38.8	305.92	671.41
1	0	TAD×18	51.7	31.15	61.72	488.92	854.53
	1	TAD×18	51.7	31.15	61.72	488.92	854.53

* Calculated as fs=fpll/2, fx/4

Table:14.3.3 The Calculation method of A/D conversion time

ANCK2	ANCK1	ANCK0	A/D conversion clock	The method of Calculation of A/D conversion time
0	0	0	fs/2	TS + 14 × TAD + 2.5 / fs
		1	fs/4	
	1	0	fs/8	
		1	fx × 2	TS + 14 × TAD + 3 / fs
1	X	X	fs/16	TS + 14 × TAD + 2.5 / fs

The setting value of “111” is prohibited for the ANCK2 to 0 flags of the A/D converter control register 0 (ANCTR0).

A/D conversion time indicated in Table:14.3.3 may be shorten up to one cycle time of A/D conversion cycle depending on phase differences between system clock and A/D conversion clock.

■ Built-in Ladder Resistor Control

The ANLADE flag of the ANCTR0 register is set to "1" to send a current to the ladder resistance for A/D conversion. When A/D conversion is stopped, the ANLADE flag of ANCTR0 register is set to "0" to save the power consumption.

■ A/D Conversion Starting Factor Setup

A/D conversion starting factor is set with the ANSTSEL1, 0 flag of the ANCTR2 register.
The ANSTSEL1, 0 flags of the ANCTR2 register is set to start an external interrupt, and A/D converter interrupt factor is set to start A/D conversion. Also, the ANST flag of the ANCTR2 register is set to "1" is possible.

When the external interrupt 2 is selected as A/D conversion starting factor, the valid edge should be assigned at REDG2 flag of the external interrupt 2 control register (IRQ2ICR) and EDGSEL1 flag of the both edge interrupt control register (EDGDT).

The interrupt valid edge assignment should be done before selecting the interrupt factor at A/D conversion starting factor.

■ A/D Conversion Starting Setup

The A/D conversion start is set with the ANST flag of the ANCTR2 register. The A/D conversion is started by setting the ANST flag of the ANCTR2 register to "1". When the A/D conversion is started by the external interrupt 2 factor, the ANST flag of the ANCTR2 register is automatically set to "1" after the external interrupt 2 is generated and the A/D conversion is started. The ANST flag of the ANCTR2 register is cleared to "0" automatically after the conversion data is stored.

14.3.2 Setup Example

■ Example of A/D Conversion Setup by Registers

A/D conversion is started by setting registers. The analog input pins are set to AN0, the conversion clock is set to fs/4, and the sampling hold time is set to T_{AD} × 2. Then, A/D conversion complete interrupt is generated.
An example setup procedure, with a description of each step is shown below.

Setup Procedure	Description
(1) Set the analog input pin. PAIMD(0x03ECA) bp0: PAIMD0 =1 PAPLU(0x03EAA) bp0: PAPLU0 =0	(1) Set the analog input pin to be set (2) to the special function pin by Port A input mode register (PAIMD) and set to the no pull-up resistor by Port A pull-up resistor control register (PAPLU).
(2) Select the analog input pin. ANCTR1(0x03FC6) bp3 to 0: ANCHS3 to 0 =000	(2) Select the analog input pin from AN0 to AN11 by setting the ANCHAS3 to 0 flags of A/D converter control register 1 (ANCTR1).
(3) Select the A/D conversion clock. ANCTR0(0x03FC5) bp2, 5 to 4: ANCK2 to 0 =001	(3) Select the A/D conversion clock by the ANCK2 to 0 flags of A/D converter control register 0 (ANCTR0).
(4) Set the sample and hold time. ANCTR0(0x03FC5) bp7 to 6: ANSH1 to 0 =00	(4) Set the sample and hold time by the ANSH1 to 0 flags of A/D converter control register 0 (ANCTR0).
(5) Set the interrupt level. ADICR(0x03FFD) bp7 to 6: ADLV1 to 0 =00	(5) Set the interrupt level by the ADLV1 to 0 flag of the A/D converter complete interrupt control register (ADICR). If any interrupt request flag is already set, clear it. [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup]
(6) Enable the interrupt. ADICR(0x03FFD) bp1: ADIE =1	(6) Enable the interrupt by setting the ADIE flag of ADICR register to "1".
(7) Set the A/D ladder resistance. ANCTR0(0x03FC5) bp3: ANLADE =1	(7) Set the ANLADE flag of A/D converter control register 0 (ANCTR0) to "1" to send a current to the ladder resistance for the A/D conversion.
(8) A/D conversion starting factor select. ANCTR2(0x03FC7) bp6 to 5: ANSTSEL1 to 0 =00	(8) Clear the ANSTSEL1 to 0 flags of A/D converter control register 2 (ANCTR2) to "00" to set A/D converter starting factor to the ANST flag of A/D converter control register 2 (ANCTR2).
(9) Start A/D converter operation. ANCTR2(0x03FC7) bp7: ANST =1	(9) Set the ANST flag of A/D converter control register 2 (ANCTR2) to "1" to start the A/D conversion.

* The above (3) to (4) can be set at the same time.

When the conversion is restarted by changing the setting after the A/D conversion, set the ANLADE flag of A/D converter control register 0 (ANCTR0) to "0" to disable the analog to change the setting.
The operation is not guaranteed if this procedure fails to be kept.
The channel switching is not considered to change the setting.

■ Example of A/D Conversion Setup by External interrupt

A/D conversion is started by the external interrupt. The analog input pins are set to AN0, the conversion clock is set to fs/4, and the sampling hold time is set to T_{AD} × 2. Then, A/D conversion complete interrupt is generated. An example setup procedure, with a description of each step is shown below.

Setup Procedure	Description
(1) Set the analog input pin. PAIMD(0x03ECA) bp0: PAIMD0 =1 PAPLU(0x03EAA) bp0: PAPLU0 =0	(1) Set the analog input pin to be set (2) to the special function pin by Port A input mode register (PAIMD) and set to the no pull-up resistor by PAPLU register.
(2) Select the analog input pin. ANCTR1(0x03FC6) bp3 to 0: ANCHS3 to 0 =0000	(2) Select the analog input pin from AN0 to AN11 by setting the ANCHAS3 to 0 flags of A/D converter control register 1 (ANCTR1).
(3) Select the A/D conversion clock. ANCTR0(0x03FC5) bp2, 5 to 4: ANCK2 to 0 =001	(3) Select the A/D conversion clock by the ANCK2 to 0 flags of A/D converter control register 0 (ANCTR0).
(4) Set the sample and hold time. ANCTR0(0x03FC5) bp7 to 6: ANSH1 to 0 =00	(4) Set the sample and hold time by the ANSH1 to 0 flags of A/D converter control register 0 (ANCTR0).
(5) Set the interrupt level. ADICR(0x03FFD) bp7 to 6: ADLV1 to 0 =00	(5) Set the interrupt level by the ADLV1 to 0 flag of A/D converter complete interrupt control register (ADICR). If any interrupt request flag is already set, clear it. [Chapter 4 4.1.4 Maskable Interrupt Control Register Setup]
(6) Enable the interrupt. ADICR(0x03FFD) bp1: ADIE =1	(6) Enable the interrupt by setting the ADIE flag of ADICR register to "1".
(7) Set the A/D ladder resistance. ANCTR0(0x03FC5) bp3: ANLADE =1	(7) Set the ANLADE flag of A/D converter control register 0 (ANCTR0) to "1" to send a current to the ladder resistance for the A/D conversion.
(8) A/D conversion starting factor select. ANCTR2(0x03FC7) bp6 to 5: ANSTSEL1 to 0 =10	(8) Set the ANSTSEL1 to 0 flags of A/D converter control register 2 (ANCTR2) to "10" to set A/D converter starting factor to the external interrupt and ANST flag of A/D converter control register 2 (ANCTR2).
(9) Start A/D converter operation. ANCTR2(0x03FC7) bp7: ANST =1	(9) When the external interrupt is generated, ANST flag of A/D converter control register 2 (ANCTR2) is set to “1” automatically, then start A/D conversion. If the external interrupt is not generated, the A/D conversion can start by setting the ANST flag of A/D converter control register 2 (ANCTR2) to "1".

When the conversion is restarted by changing the setting after the A/D conversion is completed, set the ANLADE flag of A/D converter control register 0 (ANCTR0) to "0" to disable the analog to change the setting.
The operation is not guaranteed if this procedure fails to be kept.
The channel switching is not considered to change the setting.

Input the pulse for the external interrupt to be input longer than the system clock period.
If the pulse is shorter than the system clock period, the A/D conversion may not be started.

14.3.3 A/D Control for Touch Sensor

Touch Sensor of an electric capacity system is realizable using an A/D converter.
When the TS0ADEN flag of TS0ADCNT register is set, pin TS0RC is connected to the analog channel selected by ANCTR1 register.
Refer to the application note for the detail of setup example.

Do not operate Touch Sensor Timer while the TS0ADEN flag is “1”.

The state of pin TS0RC follows the setup of Port control registers.
Set to the initial state (port selection, input mode, no pull-up or pull-down).

14.3.4 Cautions

A/D conversion can be damaged by noise easily, therefore, anti-noise measures should be taken adequately.

- Anti-noise measures

To A/D input (analog input pin), add condenser near pin VSS of microcontroller.

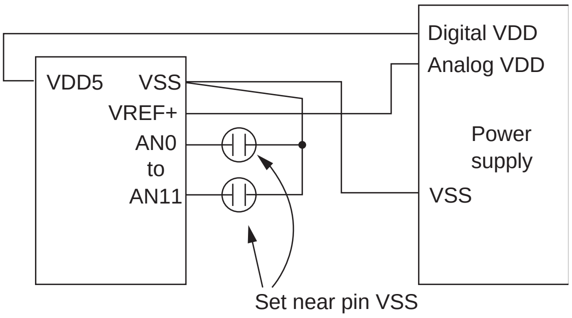


Figure:14.3.2 A/D Converter Recommended Example 1

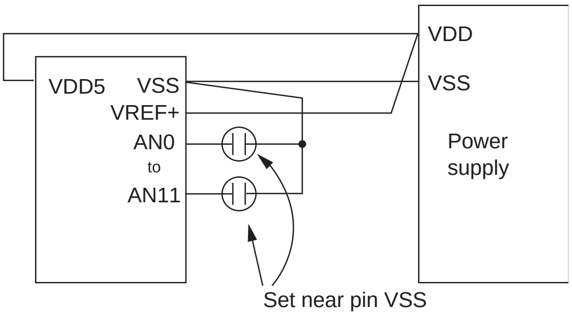


Figure:14.3.3 A/D Converter Recommended Example 2

■ Recommended Circuit with the A/D Conversion

For high precision of A/D conversion, the following cautions on A/D converter should be kept.

Set the sample hold time(T_{AD}) based on the combined capacitor of the internal equivalent circuit(C_{ad} = 12pF), the combined resistance(R_{ad} = 1900Ω), the impedance of the external analog signal output circuit(R_o) and the external capacitor(C_o).
For the sample hold time(T_{AD}), more than 7 times the value of the time constant of τ is recommended.

$$T_{AD} \geq 7 \times \tau = 7 \times (R_{ad} + R_o) \times (C_{ad} + C_o)$$

At the A/D conversion, if the output level of LSI is changed, or the peripheral added circuit is switched to ON/OFF, the input from the analog input pins and power supply pins can't be fixed. Therefore, the A/D conversion precision can't be assured. At the set checking, confirm the waveform of analog input pins.

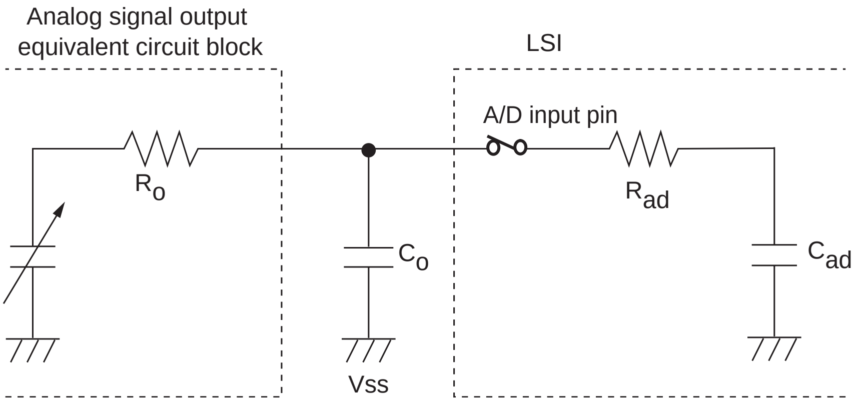


Figure:14.3.4 Circuit with the A/D Conversion

Chapter 15 Touch Sensor Timer

15.1 Overview

MN101EFA8 has 2 Touch sensor Timer for the capacitive touch sensor (Touch Sensor Timer 0 and 1).
MN101EFA7 has a Touch sensor Timer for the capacitive touch sensor (Touch Sensor Timer 0).
Touch Sensor Timer always measures the amount of change by the load capacitance of the touch sensor input pins, and generates interrupt when a measurement result satisfies the detection conditions to the expected data setup beforehand.
Touch sensor input pin can be selected maximum 12 channels (8 channels).

Table:15.1.1 Touch Sensor Channel Functions

Function		Pin Name			
		MN101EFA8	MN101EFA3	MN101EFA7	MN101EFA2
Touch Sensor 0	TS0IN0	PB0	-	P94	-
	TS0IN1	PB1	-	P57	-
	TS0IN2	PB2	-	P56	-
	TS0IN3	PB3	-	P55	-
	TS0IN4	P94	-	P54	-
	TS0IN5	P93	-	P53	-
	TS0IN6	P92	-	P52	-
	TS0IN7	P33	-	P51	-
Touch Sensor 1	TS1IN0	P45	-	-	-
	TS1IN1	P46	-	-	-
	TS1IN2	P47	-	-	-
	TS1IN3	P57	-	-	-

15.1.1 Functions

Table:15.1.2 shows the Touch Sensor Timer 0 and 1 functions.

Table:15.1.2 Touch Sensor Timer 0 and 1 Functions

Functions	Contents	
	Touch Sensor Timer 0	Touch Sensor Timer 1
Interrupts operation	The interrupt is generated when a measurement result satisfies the detection conditions by comparing with the expected data after the input pulse from touch sensor input pin is counted.	
Touch Detection Mode	RC mode OSC mode	
Touch Sensor Input Channel	TS0IN0 to TS0IN7	TS1IN0 to TS1IN3
	maximum 8 channels	maximum 4 channels
Interrupts	TS0DTIRQ	TS1DTIRQ
	TS0DEIRQ	TS1DEIRQ
	TS0CIRQ	TS1CIRQ
	TS0ATIRQ	TS1ATIRQ
Clock Source	fpll-div/1, 2, 4, 8, 16, 32	
Count Mode	Channel cycle mode 1: Detection operation is executed in order of channel number. After detection operation takes a round of channels, it stops.	
	Channel cycle mode 2: Cyclic detection operation is continued in order of channel number.	
Detection Resistor	Internal pull-up/pull-down resistor External resistor	
Data Automatic Transfer	The value of measurement result registers can be transferred to the arbitrary address in RAM by occurring each interrupt of touch sensor timer as a trigger factor.	

15.1.2 Block Diagram

■ Touch Sensor Timer 0 Block Diagram

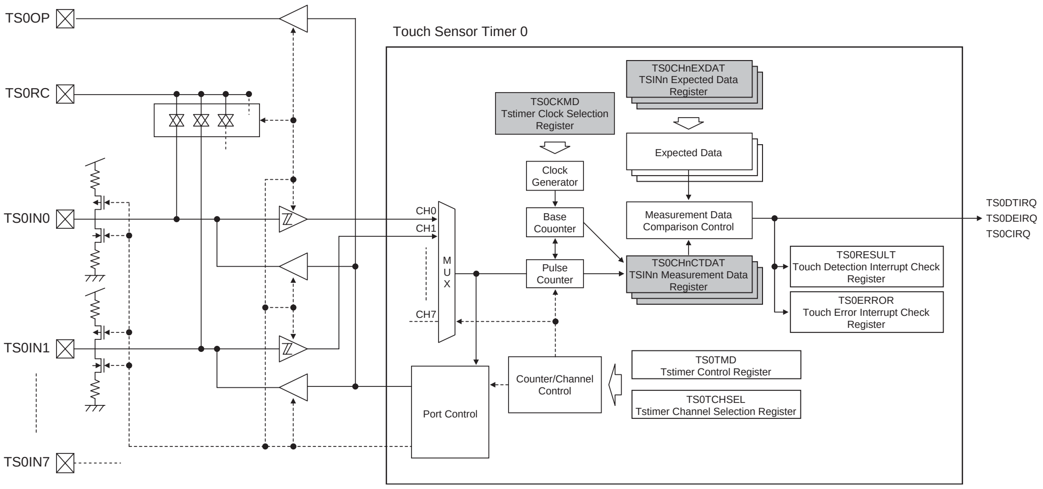


Figure:15.1.1 Touch Sensor Timer 0 Block Diagram

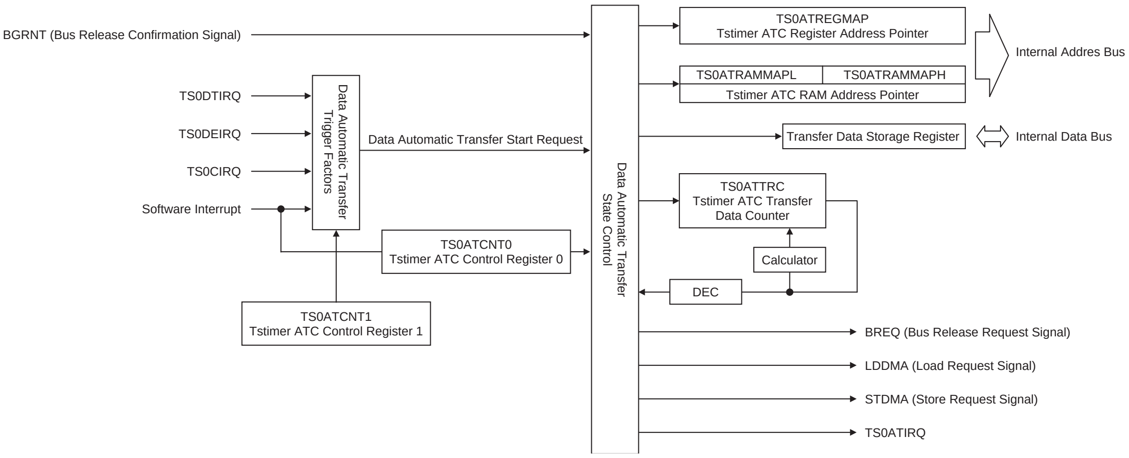

BGRNT (Bus Release Confirmation Signal)
TS0DTIRQ
TS0DEIRQ
TS0CIRQ
Software Interrupt
Data Automatic Transfer Trigger Factors
Data Automatic Transfer Start Request
TS0ATCNT0
Tstimer ATC Control Register 0
TS0ATCNT1
Tstimer ATC Control Register 1
Data Automatic Transfer State Control
TS0ATREGMAP
Tstimer ATC Register Address Pointer
Internal Addres Bus
TS0ATRAMMAPL
TS0ATRAMMAPH
Tstimer ATC RAM Address Pointer
Transfer Data Storage Register
Internal Data Bus
TS0ATTRC
Tstimer ATC Transfer Data Counter
Calculator
DEC
BREQ (Bus Release Request Signal)
LDDMA (Load Request Signal)
STDMA (Store Request Signal)
TS0ATIRQ

Figure:15.1.2 Data Automatic Transfer 0 Block Diagram

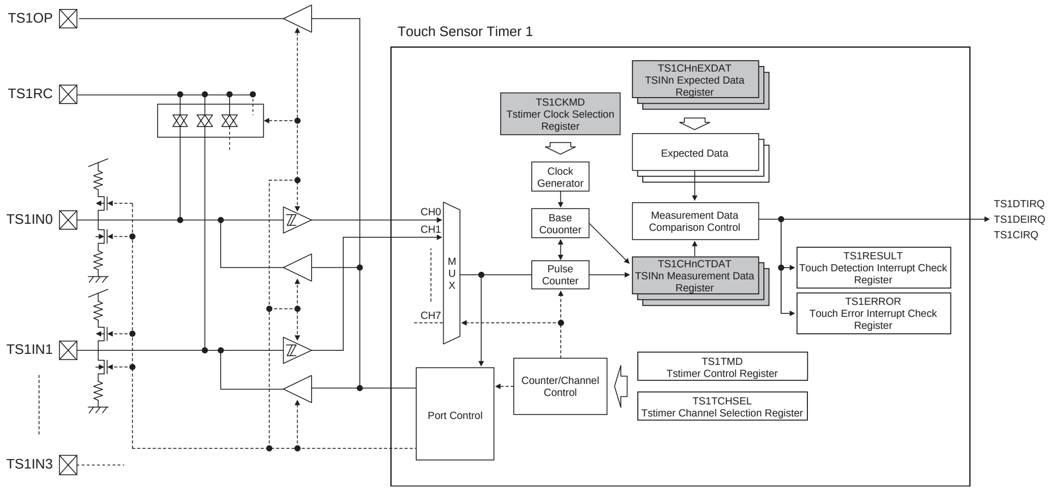


Figure:15.1.3 Touch Sensor Timer 1 Block Diagram (only MN101EFA8)

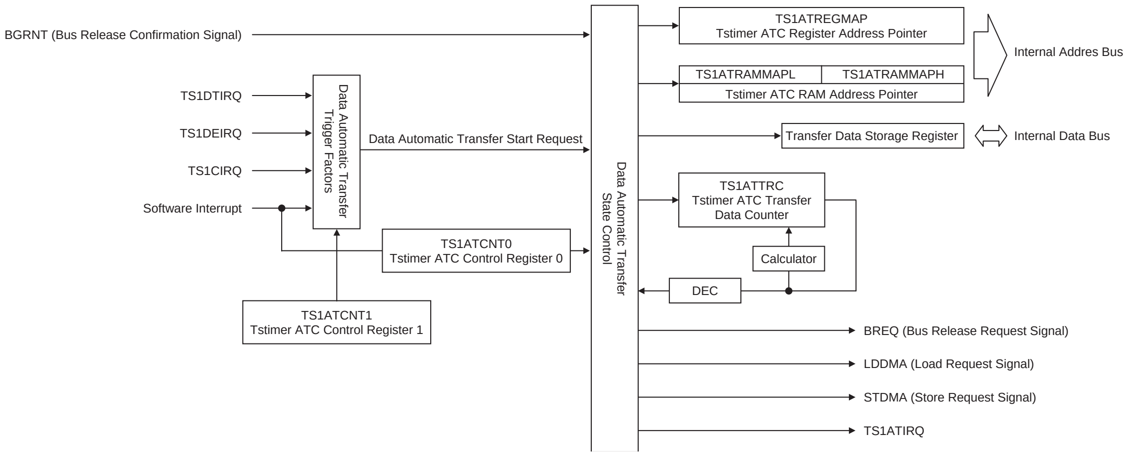

BGRNT (Bus Release Confirmation Signal)
TS1DTIRQ
TS1DEIRQ
TS1CIRQ
Software Interrupt
Data Automatic Transfer
Trigger Factors
Data Automatic Transfer Start Request
TS1ATCNT0
Tstimer ATC Control Register 0
TS1ATCNT1
Tstimer ATC Control Register 1
Data Automatic Transfer
State Control
TS1ATREGMAP
Tstimer ATC Register Address Pointer
Internal Addres Bus
TS1ATRAMMAPL
TS1ATRAMMAPH
Tstimer ATC RAM Address Pointer
Transfer Data Storage Register
Internal Data Bus
TS1ATTRC
Tstimer ATC Transfer
Data Counter
Calculator
DEC
BREQ (Bus Release Request Signal)
LDDMA (Load Request Signal)
STDMA (Store Request Signal)
TS1ATIRQ

Figure:15.1.4 Data Automatic Transfer 1 Block Diagram (only MN101EFA8)

15.2 Control Registers

15.2.1 Registers

Table:15.2.1 shows the registers used to control Touch sensor timer.

Table:15.2.1 Touch Sensor Timer Control Registers

Register		Address	R/W	Function	Page	MN101 EFA8	MN101 EFA3	MN101 EFA7	MN101 EFA2
Touch Sensor Timer 0	TS0TMD	0x03DA0	R/W	Tstimer 0 control register	XV-9	√	-	√	-
	TS0CKMD	0x03DA1	R/W	Tstimer 0 clock selection register	XV-11	√	-	√	-
	TS0TCHSEL	0x03DA2	R/W	Tstimer 0 channel selection register	XV-12	√	-	√	-
	TS0RESULT	0x03DA3	R/W	Touch 0 detect interrupt check register	XV-14	√	-	√	-
	TS0ERROR	0x03DA4	R/W	Touch 0 error interrupt check register	XV-16	√	-	√	-
	TS0CH0EXDATL	0x03DB0	R/W	TS0IN0 expected data register (lower 8 bits)	XV-18	√	-	√	-
	TS0CH0EXDATH	0x03DB1	R/W	TS0IN0 expected data register (upper 2 bits)	XV-18	√	-	√	-
	TS0CH1EXDATL	0x03DB2	R/W	TS0IN1 expected data register (lower 8 bits)	XV-18	√	-	√	-
	TS0CH1EXDATH	0x03DB3	R/W	TS0IN1 expected data register (upper 2 bits)	XV-18	√	-	√	-
	TS0CH2EXDATL	0x03DB4	R/W	TS0IN2 expected data register (lower 8 bits)	XV-18	√	-	√	-
	TS0CH2EXDATH	0x03DB5	R/W	TS0IN2 expected data register (upper 2 bits)	XV-18	√	-	√	-
	TS0CH3EXDATL	0x03DB6	R/W	TS0IN3 expected data register (lower 8 bits)	XV-18	√	-	√	-
	TS0CH3EXDATH	0x03DB7	R/W	TS0IN3 expected data register (upper 2 bits)	XV-18	√	-	√	-
	TS0CH4EXDATL	0x03DB8	R/W	TS0IN4 expected data register (lower 8 bits)	XV-18	√	-	√	-
	TS0CH4EXDATH	0x03DB9	R/W	TS0IN4 expected data register (upper 2 bits)	XV-18	√	-	√	-
	TS0CH5EXDATL	0x03DBA	R/W	TS0IN5 expected data register (lower 8 bits)	XV-18	√	-	√	-
	TS0CH5EXDATH	0x03DBB	R/W	TS0IN5 expected data register (upper 2 bits)	XV-18	√	-	√	-
	TS0CH6EXDATL	0x03DBC	R/W	TS0IN6 expected data register (lower 8 bits)	XV-18	√	-	√	-
	TS0CH6EXDATH	0x03DBD	R/W	TS0IN6 expected data register (upper 2 bits)	XV-18	√	-	√	-
	TS0CH7EXDATL	0x03DBE	R/W	TS0IN7 expected data register (lower 8 bits)	XV-18	√	-	√	-
	TS0CH7EXDATH	0x03DBF	R/W	TS0IN7 expected data register (upper 2 bits)	XV-18	√	-	√	-

Register		Address	R/W	Function	Page	MN101 EFA8	MN101 EFA3	MN101 EFA7	MN101 EFA2
Touch Sensor Timer 0	TS0CH0CTDATL	0x03DC0	R	TS0IN0 measurement data register (lower 8 bits)	XV-19	√	-	√	-
	TS0CH0CTDATH	0x03DC1	R	TS0IN0 measurement data register (upper 2 bits)	XV-19	√	-	√	-
	TS0CH1CTDATL	0x03DC2	R	TS0IN1 measurement data register (lower 8 bits)	XV-19	√	-	√	-
	TS0CH1CTDATH	0x03DC3	R	TS0IN1 measurement data register (upper 2 bits)	XV-19	√	-	√	-
	TS0CH2CTDATL	0x03DC4	R	TS0IN2 measurement data register (lower 8 bits)	XV-19	√	-	√	-
	TS0CH2CTDATH	0x03DC5	R	TS0IN2 measurement data register (upper 2 bits)	XV-19	√	-	√	-
	TS0CH3CTDATL	0x03DC6	R	TS0IN3 measurement data register (lower 8 bits)	XV-19	√	-	√	-
	TS0CH3CTDATH	0x03DC7	R	TS0IN3 measurement data register (upper 2 bits)	XV-19	√	-	√	-
	TS0CH4CTDATL	0x03DC8	R	TS0IN4 measurement data register (lower 8 bits)	XV-19	√	-	√	-
	TS0CH4CTDATH	0x03DC9	R	TS0IN4 measurement data register (upper 2 bits)	XV-19	√	-	√	-
	TS0CH5CTDATL	0x03DCA	R	TS0IN5 measurement data register (lower 8 bits)	XV-19	√	-	√	-
	TS0CH5CTDATH	0x03DCB	R	TS0IN5 measurement data register (upper 2 bits)	XV-19	√	-	√	-
	TS0CH6CTDATL	0x03DCC	R	TS0IN6 measurement data register (lower 8 bits)	XV-19	√	-	√	-
	TS0CH6CTDATH	0x03DCD	R	TS0IN6 measurement data register (upper 2 bits)	XV-19	√	-	√	-
	TS0CH7CTDATL	0x03DCE	R	TS0IN7 measurement data register (lower 8 bits)	XV-19	√	-	√	-
	TS0CH7CTDATH	0x03DCF	R	TS0IN7 measurement data register (upper 2 bits)	XV-19	√	-	√	-
	TS0ATCNT0	0x03E60	R/W	Tstimer ATC 0 control register 0	XV-20	√	-	√	-
	TS0ATCNT1	0x03E61	R/W	Tstimer ATC 0 control register 1	XV-21	√	-	√	-
	TS0ATTRC	0x03E62	R/W	Tstimer ATC 0 transfer data counter	XV-21	√	-	√	-
	TS0ATRAMAPL	0x03E63	R/W	Tstimer ATC 0 RAM address pointer (lower 8 bits)	XV-22	√	-	√	-
	TS0ATRAMAPH	0x03E64	R/W	Tstimer ATC 0 RAM address pointer (upper 5 bits)	XV-22	√	-	√	-
	TS0ATREGAP	0x03E66	R/W	Tstimer ATC 0 register address pointer	XV-23	√	-	√	-

Register		Address	R/W	Function	Page	MN101 EFA8	MN101 EFA3	MN101 EFA7	MN101 EFA2
Touch Sensor Timer 1	TS1TMD	0x03DD0	R/W	Tstimer 1 control register	XV-9	√	-	-	-
	TS1CKMD	0x03DD1	R/W	Tstimer 1 clock selection register	XV-11	√	-	-	-
	TS1TCHSEL	0x03DD2	R/W	Tstimer 1 channel selection register	XV-12	√	-	-	-
	TS1RESULT	0x03DD3	R/W	Touch 1 detect interrupt check register	XV-14	√	-	-	-
	TS1ERROR	0x03DD4	R/W	Touch 1 error interrupt check register	XV-16	√	-	-	-
	TS1CH0EXDATL	0x03DE0	R/W	TS1IN0 expected data register (lower 8 bits)	XV-18	√	-	-	-
	TS1CH0EXDATH	0x03DE1	R/W	TS1IN0 expected data register (upper 2 bits)	XV-18	√	-	-	-
	TS1CH1EXDATL	0x03DE2	R/W	TS1IN1 expected data register (lower 8 bits)	XV-18	√	-	-	-
	TS1CH1EXDATH	0x03DE3	R/W	TS1IN1 expected data register (upper 2 bits)	XV-18	√	-	-	-
	TS1CH2EXDATL	0x03DE4	R/W	TS1IN2 expected data register (lower 8 bits)	XV-18	√	-	-	-
	TS1CH2EXDATH	0x03DE5	R/W	TS1IN2 expected data register (upper 2 bits)	XV-18	√	-	-	-
	TS1CH3EXDATL	0x03DE6	R/W	TS1IN3 expected data register (lower 8 bits)	XV-18	√	-	-	-
	TS1CH3EXDATH	0x03DE7	R/W	TS1IN3 expected data register (upper 2 bits)	XV-18	√	-	-	-
	TS1CH0CTDATL	0x03DF0	R	TS1IN0 measurement data register (lower 8 bits)	XV-19	√	-	-	-
	TS1CH0CTDATH	0x03DF1	R	TS1IN0 measurement data register (upper 2 bits)	XV-19	√	-	-	-
	TS1CH1CTDATL	0x03DF2	R	TS1IN1 measurement data register (lower 8 bits)	XV-19	√	-	-	-
	TS1CH1CTDATH	0x03DF3	R	TS1IN1 measurement data register (upper 2 bits)	XV-19	√	-	-	-
	TS1CH2CTDATL	0x03DF4	R	TS1IN2 measurement data register (lower 8 bits)	XV-19	√	-	-	-
	TS1CH2CTDATH	0x03DF5	R	TS1IN2 measurement data register (upper 2 bits)	XV-19	√	-	-	-
	TS1CH3CTDATL	0x03DF6	R	TS1IN3 measurement data register (lower 8 bits)	XV-19	√	-	-	-
	TS1CH3CTDATH	0x03DF7	R	TS1IN3 measurement data register (upper 2 bits)	XV-19	√	-	-	-
	TS1ATCNT0	0x03E68	R/W	Tstimer ATC 1 control register 0	XV-20	√	-	-	-
	TS1ATCNT1	0x03E69	R/W	Tstimer ATC 1 control register 1	XV-21	√	-	-	-
	TS1ATTRC	0x03E6A	R/W	Tstimer ATC 1 transfer data counter	XV-21	√	-	-	-
	TS1ATRAMAPL	0x03E6B	R/W	Tstimer ATC 1 RAM address pointer (lower 8 bits)	XV-22	√	-	-	-
	TS1ATRAMAPH	0x03E6C	R/W	Tstimer ATC 1 RAM address pointer (upper 5 bits)	XV-22	√	-	-	-
	TS1ATREGAP	0x03E6E	R/W	Tstimer ATC 1 register address pointer	XV-23	√	-	-	-

R/W: Readable/Writable
R: Read only

15.2.2 Control Registers

In this section, “n” is described as a series of numbers from 0 to 1.

■ Tstimer n Control Register (TS0TMD: 0x03DA0, TS1TMD: 0x03DD0)

Tstimer n control register controls detection mode, selection of detection resistors and setting to enable or disable of the timer counting.

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	TSnCSMD	TSnRSMD	TSnRCMD	TSnMD	TSnSTEN
At Reset	-	-	-	0	0	0	0	0
Access	-	-	-	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7-5	-	-
4	TSnCSMD	Channel cycle mode selection 0: Channel mode 1 1: Channel mode 2
3	TSnRSMD	Detection resistor selection 0: Internal resistor 1: External resistor
2	TSnRCMD	RC mode selection 0: Charge 1: Discharge
1	TSnMD	Detection mode selection 0: RC mode 1: OSC mode
0	TSnSTEN	Timer count control 0: Disabled 1: Enabled

Set the TSnSTEN flag to “1” after setting up the other flags.

Be sure to set the TSnRCMD flag to “0”, when the TSnMD flag is “1”.

When the TSnCSMD flag is “0”, touch sensor timer stops automatically and the TSnSTEN flag is cleared to “0”, after detection operation executes a round of channels.
When the TSnCSMD flag is “1”, touch sensor timer continues counting until the TSnSTEN flag is set to “0”.

■ Tstimer n Clock Selection Register (TS0CKMD: 0x03DA1, TS1CKMD: 0x03DD1)

Tstimer n clock selection register selects the clock source of base counter.

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	TSnCK2	TSnCK1	TSnCK0	TSnPSCEN
At Reset	-	-	-	-	0	0	0	0
Access	-	-	-	-	R/W	R/W	R/W	R/W

bp	Flag	Description
7-4	-	-
3-1	TSnCK2-0	Clock source selection 000: fpll-div 001: fpll-div/2 010: fpll-div/4 011: fpll-div/8 100: fpll-div/16 101: fpll-div/32 110: Setting prohibited 111: Setting prohibited
0	TSnPSCEN	Clock source output control 0: Disabled 1: Enabled

Set the TSnPSCEN flag to “1” after setting the TSnCK2 to TSnCK0 flags.

Do not change the value of TSnCK2 to TSnCK0 flags while the TSnPSCEN flag is “1”.

■ Tstimer n Channel Selection Register (TS0TCHSEL: 0x03DA2, TS1TCHSEL: 0x03DD2)

Tstimer n channel selection register selects the channel used as the input pin for touch sensor.

MN101EFA8/A7

bp	7	6	5	4	3	2	1	0
Flag	TS0IN7 SEL	TS0IN6 SEL	TS0IN5 SEL	TS0IN4 SEL	TS0IN3 SEL	TS0IN2 SEL	TS0IN1 SEL	TS0IN0 SEL
At Reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7	TS0IN7SEL	TS0IN7 channel selection 0: Unused 1: Used
6	TS0IN6SEL	TS0IN6 channel selection 0: Unused 1: Used
5	TS0IN5SEL	TS0IN5 channel selection 0: Unused 1: Used
4	TS0IN4SEL	TS0IN4 channel selection 0: Unused 1: Used
3	TS0IN3SEL	TS0IN3 channel selection 0: Unused 1: Used
2	TS0IN2SEL	TS0IN2 channel selection 0: Unused 1: Used
1	TS0IN1SEL	TS0IN1 channel selection 0: Unused 1: Used
0	TS0IN0SEL	TS0IN0 channel selection 0: Unused 1: Used

MN101EFA8

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	TS1IN3 SEL	TS1IN2 SEL	TS1IN1 SEL	TS1IN0 SEL
At Reset	-	-	-	-	0	0	0	0
Access	-	-	-	-	R/W	R/W	R/W	R/W

bp	Flag	Description
7-4	-	-
3	TS1IN3SEL	TS1IN3 channel selection 0: Unused 1: Used
2	TS1IN2SEL	TS1IN2 channel selection 0: Unused 1: Used
1	TS1IN1SEL	TS1IN1 channel selection 0: Unused 1: Used
0	TS1IN0SEL	TS1IN0 channel selection 0: Unused 1: Used

While each flag is set to “1”, the corresponding pin is used as the channel of Touch Sensor Timer n regardless of the setting of port control registers.

Touch Sensor Timer n executes the detection operation sequentially from CH0 for TSnINm set to “Used”, and skips TSnINm set to “Unused”.

Standard Operation:

CH0 → CH1 → CH2 → CH3 → CH4 → CH5 → CH6 → CH7 → CH0 → CH1....

When the TSnIN3SEL and TSnIN5SEL flags are set to “0”:

CH0 → CH1 → CH2 → CH4 → CH6 → CH7 → CH0 → CH1....

Touch Sensor Timer n generates a tstimer channel selection interrupt when the detection operation is completed to a cycle of the channels set to “Used”.
When the TSnCSMD flag of TSnTMD register is “0”, the detection operation is halted and a tstimer channel selection interrupt is generated.
When the TSnCSMD flag of TSnTMD register is “1”, the detection operation is continued and tstimer channel selection interrupt is generated every a cycle.

■ Touch n Detect Interrupt Check Register (TS0RESULT: 0x03DA3, TS1RESULT: 0x03DD3)

Touch n Detect interrupt check register confirms the channel generated touch detection interrupt.

MN101EFA8/A7

bp	7	6	5	4	3	2	1	0
Flag	TS0RCH7	TS0RCH6	TS0RCH5	TS0RCH4	TS0RCH3	TS0RCH2	TS0RCH1	TS0RCH0
At Reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7	TS0RCH7	TS0IN7 Touch detect interrupt 0: No interrupt has been generated 1: An interrupt has been generated
6	TS0RCH6	TS0IN6 Touch detect interrupt 0: No interrupt has been generated 1: An interrupt has been generated
5	TS0RCH5	TS0IN5 Touch detect interrupt 0: No interrupt has been generated 1: An interrupt has been generated
4	TS0RCH4	TS0IN4 Touch detect interrupt 0: No interrupt has been generated 1: An interrupt has been generated
3	TS0RCH3	TS0IN3 Touch detect interrupt 0: No interrupt has been generated 1: An interrupt has been generated
2	TS0RCH2	TS0IN2 Touch detect interrupt 0: No interrupt has been generated 1: An interrupt has been generated
1	TS0RCH1	TS0IN1 Touch detect interrupt 0: No interrupt has been generated 1: An interrupt has been generated
0	TS0RCH0	TS0IN0 Touch detect interrupt 0: No interrupt has been generated 1: An interrupt has been generated

MN101EFA8

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	TS1RCH3	TS1RCH2	TS1RCH1	TS1RCH0
At Reset	-	-	-	-	0	0	0	0
Access	-	-	-	-	R/W	R/W	R/W	R/W

bp	Flag	Description
7-4	-	-
3	TS1RCH3	TS1IN3 Touch detect interrupt 0: No interrupt has been generated 1: An interrupt has been generated
2	TS1RCH2	TS1IN2 Touch detect interrupt 0: No interrupt has been generated 1: An interrupt has been generated
1	TS1RCH1	TS1IN1 Touch detect interrupt 0: No interrupt has been generated 1: An interrupt has been generated
0	TS1RCH0	TS1IN0 Touch detect interrupt 0: No interrupt has been generated 1: An interrupt has been generated

Touch Sensor Timer n continues executing the detection operation, even if a Touch n detect interrupt is occurred.
Therefore, Touch n detect interrupt may have been generated in two or more channels depend on the timing to read the TSnRESULT register.

While each bit of the TSnRESULT register is “0”, “1” cannot be written by software.
When each bit of the TSnRESULT register is “1”, “0” can be set by writing to “1”.
Each bit once set to “1” is not set to “0” by hardware, even if corresponding channel is not detected in subsequent operations.

■ Touch n Detect Error Interrupt Check Register (TS0ERROR: 0x03DA4, TS1ERROR: 0x03DD4)

Touch n Detect Error interrupt check register confirms the channel generated Touch n detect error interrupt.

MN101EFA8/A7

bp	7	6	5	4	3	2	1	0
Flag	TS0ECH7	TS0ECH6	TS0ECH5	TS0ECH4	TS0ECH3	TS0ECH2	TS0ECH1	TS0ECH0
At Reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

bp	Flag	Description
7	TS0ECH7	TS0IN7 Touch detect error interrupt 0: No interrupt has been generated 1: An interrupt has been generated
6	TS0ECH6	TS0IN6 Touch detect error interrupt 0: No interrupt has been generated 1: An interrupt has been generated
5	TS0ECH5	TS0IN5 Touch detect error interrupt 0: No interrupt has been generated 1: An interrupt has been generated
4	TS0ECH4	TS0IN4 Touch detect error interrupt 0: No interrupt has been generated 1: An interrupt has been generated
3	TS0ECH3	TS0IN3 Touch detect error interrupt 0: No interrupt has been generated 1: An interrupt has been generated
2	TS0ECH2	TS0IN2 Touch detect error interrupt 0: No interrupt has been generated 1: An interrupt has been generated
1	TS0ECH1	TS0IN1 Touch detect error interrupt 0: No interrupt has been generated 1: An interrupt has been generated
0	TS0ECH0	TS0IN0 Touch detect error interrupt 0: No interrupt has been generated 1: An interrupt has been generated

MN101EFA8

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	TS1ECH3	TS1ECH2	TS1ECH1	TS1ECH0
At Reset	-	-	-	-	0	0	0	0
Access	-	-	-	-	R/W	R/W	R/W	R/W

bp	Flag	Description
7-4	-	-
3	TS1ECH3	TS1IN3 Touch detect error interrupt 0: No interrupt has been generated 1: An interrupt has been generated
2	TS1ECH2	TS1IN2 Touch detect error interrupt 0: No interrupt has been generated 1: An interrupt has been generated
1	TS1ECH1	TS1IN1 Touch detect error interrupt 0: No interrupt has been generated 1: An interrupt has been generated
0	TS1ECH0	TS1IN0 Touch detect error interrupt 0: No interrupt has been generated 1: An interrupt has been generated

Touch Sensor Timer continues executing the detection operation, even if a Touch detect error interrupt is occurred.
Therefore, Touch detect error interrupt may have been generated in two or more channels depend on the timing to read TSnERROR register.

While each bit of the TSnERROR register is “0”, “1” cannot be written by software.
When each bit of the TSnERROR register is “1”, “0” can be set by writing to “1”.
Each bit once set to “1” is not set to “0” by hardware, even if corresponding channel is not detected in subsequent operations.

■ TSnINm Expected Data Register (Lower 8 bits)
(TS0CH0EXDATL: 0x03DB0, TS0CH1EXDATL: 0x03DB2, TS0CH2EXDATL: 0x03DB4,
TS0CH3EXDATL: 0x03DB6, TS0CH4EXDATL: 0x03DB8, TS0CH5EXDATL: 0x03DBA,
TS0CH6EXDATL: 0x03DBC, TS0CH7EXDATL: 0x03DBE, TS1CH0EXDATL: 0x03DE0,
TS1CH1EXDATL: 0x03DE2, TS1CH2EXDATL: 0x03DE4, TS1CH3EXDATL: 0x03DE6)

TSINn expected data register (lower 8 bits) sets the lower 8 bits of expected count data as the condition of touch detection for each channel.

bp	7	6	5	4	3	2	1	0
Flag	TSnCHm EXD7	TSnCHm EXD6	TSnCHm EXD5	TSnCHm EXD4	TSnCHm EXD3	TSnCHm EXD2	TSnCHm EXD1	TSnCHm EXD0
At Reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

■ TSnINm Expected Data Register (Upper 2 bits)
(TS0CH0EXDATH: 0x03DB1, TS0CH1EXDATH: 0x03DB3, TS0CH2EXDATH: 0x03DB5,
TS0CH3EXDATH: 0x03DB7, TS0CH4EXDATH: 0x03DB9, TS0CH5EXDATH: 0x03DBB,
TS0CH6EXDATH: 0x03DBD, TS0CH7EXDATH: 0x03DBF, TS1CH0EXDATH: 0x03DE1,
TS1CH1EXDATH: 0x03DE3, TS1CH2EXDATH: 0x03DE5, TS1CH3EXDATH: 0x03DE7)

TSnINm expected data register (upper 2 bits) sets the upper 2 bits of expected count data as the condition of touch detection for each channel.

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	-	-	TSnCHm EXD9	TSnCHm EXD8
At Reset	-	-	-	-	-	-	0	0
Access	-	-	-	-	-	-	R/W	R/W

While the TSnSTEN flag of TSnTMD register is “0”, data set in these registers are immediately reflected in the circuit.
While the TSnSTEN flag of TSnTMD register is “1”, 10 bits of data set in these registers are simultaneously reflected in the circuit, after detection operation of the channel which set the expected data is completed.

- TSnINm Measurement Data Register (Lower 8 bits)
  (TS0CH0CTDATL: 0x03DC0, TS0CH1CTDATL: 0x03DC2, TS0CH2CTDATL: 0x03DC4,
  TS0CH3CTDATL: 0x03DC6, TS0CH4CTDATL: 0x03DC8, TS0CH5CTDATL: 0x03DCA,
  TS0CH6CTDATL: 0x03DCC, TS0CH7CTDATL: 0x03DCE, TS1CH0CTDATL: 0x03DF0,
  TS1CH1CTDATL: 0x03DF2, TS1CH2CTDATL: 0x03DF4, TS1CH3CTDATL: 0x03DF6)

TSnINm measurement data register (lower 8 bits) sets the lower 8 bits of measurement data of touch detection for each channel.

bp	7	6	5	4	3	2	1	0
Flag	TSnCHm CTD7	TSnCHm CTD6	TSnCHm CTD5	TSnCHm CTD4	TSnCHm CTD3	TSnCHm CTD2	TSnCHm CTD1	TSnCHm CTD0
At Reset	0	0	0	0	0	0	0	0
Access	R	R	R	R	R	R	R	R

- TSnINm Measurement Data Register (Upper 2 bits)
  (TS0CH0CTDATH: 0x03DC1, TS0CH1CTDATH: 0x03DC3, TS0CH2CTDATH: 0x03DC5,
  TS0CH3CTDATH: 0x03DC7, TS0CH4CTDATH: 0x03DC9, TS0CH5CTDATH: 0x03DCB,
  TS0CH6CTDATH: 0x03DCD, TS0CH7CTDATH: 0x03DCF, TS1CH0CTDATH: 0x03DF1,
  TS1CH1CTDATH: 0x03DF3, TS1CH2CTDATH: 0x03DF5, TS1CH3CTDATH: 0x03DF7)

TSnINm measurement data register (upper 2 bits) sets the upper 2 bits of measurement data of touch detection for each channel.

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	-	-	-	TSnCHm CTD9	TSnCHm CTD8
At Reset	-	-	-	-	-	-	0	0
Access	-	-	-	-	-	-	R	R

■ Tstimer ATC n Control Register 0 (TS0ATCNT0: 0x03E60, TS1ATCNT0: 0x03E68)

Tstimer ATC n control register 0 controls transmitting the data of registers in touch sensor timer to arbitrary RAM address without CPU.

bp	7	6	5	4	3	2	1	0
Flag	TSnRAMDE	TSnATACT	-	-	-	-	Reserved	TSnATEN
At Reset	0	0	-	-	-	-	0	0
Access	R/W	R/W	-	-	-	-	R/W	R/W

bp	Flag	Description
7	TSnRAMDE	Increment control for RAM address pointer 0: Enabled 1: Disabled
6	TSnATACT	ATC software activation 0: Stop 1: Start
5-2	-	-
1	Reserved	Always set to “0”.
0	TSnATEN	ATC enable flag 0: Disabled 1: Enabled

The TSnATACT flag of TSnATCNT0 register is cleared to "0" by hardware automatically when the transmission is completed.

■ Tstimer ATC n Control Register 1 (TS0ATCNT1: 0x03E61, TS1ATCNT1: 0x03E69)

Tstimer ATC 0 control register 1 controls transmitting the data of registers in touch sensor timer to arbitrary RAM address without CPU.

bp	7	6	5	4	3	2	1	0
Flag	-	-	Reserved	Reserved	Reserved	Reserved	TSnATIR1	TSnATIR0
At Reset	-	-	0	0	0	0	0	0
Access	-	-	R/W	R/W	R	R	R/W	R/W

bp	Flag	Description
7-6	-	-
5-2	Reserved	Always set to “0”.
1-0	TSnATIR1-0	ATC trigger factor 00: Touch n detect interrupt 01: Touch n detect error interrupt 10: Touch n round channel selection interrupt 11: Software interrupt

Bit 5 of TSnATCNT1 register may be set to “1” by hardware automatically.
Data automatic transfer function operates properly even if bit 5 is set to “1”.

■ Tstimer ATC n Transfer Data Counter (TS0ATTRC: 0x03E62, TS1ATTRC: 0x03E6A)

Tstimer ATC n transfer data counter sets the number of registers which is transferred from the register address specified by TSnATREGAP register.

bp	7	6	5	4	3	2	1	0
Flag	TSnAT TRC7	TSnAT TRC6	TSnAT TRC5	TSnAT TRC4	TSnAT TRC3	TSnAT TRC2	TSnAT TRC1	TSnAT TRC0
At Reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

The data of TSnATTRC register is decremented during transmission operation, and when the data reaches “0x00”, the transmission is completed.
To restart the data automatic transfer, set the data to TSnATTRC register again.

- Tstimer n ATC RAM Address Pointer
  (Lower 8 bits TS0ATRAMAPL: 0x03E63, TS1ATRAMAPL: 0x03E6B)

Tstimer ATC RAM address pointer (lower 8 bits) specifies the destination start address (lower 8 bits) of RAM.

bp	7	6	5	4	3	2	1	0
Flag	TSnAT RAMAP7	TSnAT RAMAP6	TSnAT RAMAP5	TSnAT RAMAP4	TSnAT RAMAP3	TSnAT RAMAP2	TSnAT RAMAP1	TSnAT RAMAP0
At Reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- Tstimer ATC n RAM Address Pointer
  (Upper 5 bits TS0ATRAMAPH: 0x03E64, TS1ATRAMAPH: 0x03E6C)

Tstimer ATC n RAM address pointer (upper 5 bits) specifies the destination start address (upper 5 bits) of RAM.

bp	7	6	5	4	3	2	1	0
Flag	-	-	-	TSnAT RAMAP12	TSnAT RAMAP11	TSnAT RAMAP10	TSnAT RAMAP9	TSnAT RAMAP8
At Reset	-	-	-	0	0	0	0	0
Access	-	-	-	R	R	R	R/W	R/W

The data of tstimer ATC n RAM address pointers is incremented during transmission operation, and when the transmission is completed, the data is one address higher than the last transmitted address.
To restart the data automatic transfer in the same start address, set TSnATRAMAPL and TSnATRAMAPH registers again.

■ Tstimer ATC n Register Address Pointer (TS0ATREGAP: 0x03E66, TS1ATREGAP: 0x03E6E)

Tstimer ATC n register address pointer specifies the source start address (lower 8 bits) of register which is transmitted.

bp	7	6	5	4	3	2	1	0
Flag	TSnAT REGAP7	TSnAT REGAP6	TSnAT REGAP5	TSnAT REGAP4	TSnAT REGAP3	TSnAT REGAP2	TSnAT REGAP1	TSnAT REGAP0
At Reset	0	0	0	0	0	0	0	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

TSnATREGAP register specifies the lower 8 bits of the register address.
The upper data is fixed to “0x03D”.

The data of TSnATREGAP register is incremented during transmission operation, and when the transmission is completed, the data is one address higher than the last transmitted address.
To restart the data automatic transfer in the same start address, set TSnATREGAP register again.

15.3 Operation (Touch Sensor Timer)

15.3.1 Detection Mode Selection

Touch Sensor Timer 0 and 1 can be selected 2 detection modes.
This timer executes the detection operation by the base counter and the pulse counter.
Each counter operates differently depending on the detection mode.

■ RC mode

In this mode, the base counter counts the period until the polarity of touch sensor input pin changes (from "Low" level to "High" level, or from "High" level to "Low" level).
When the polarity of touch sensor input pin changes, the data of the base counter is stored in TSnCHmCTDATH and TSnCHmCTDATL registers, and the detection operation in next channel starts.

■ OSC mode

In this mode, the pulse counter counts the number of pulses input to touch sensor input pin during the cycle setup in base counter.
When the base counter overflows, the data of the pulse counter equivalent to 1 cycle of base counter is stored in TSnCH0CTDATH and TSnCHmCTDATL registers, and the detection operation in next channel starts.

Input channel of Touch Sensor Timer 0 can be selected maximum 8 channels.
Input channel of Touch Sensor Timer 1 can be selected maximum 4 channels.
The detection operation is executed sequentially from the smaller number in selected channels
When the detection operation is completed to all the selected channels, Touch 0 round interrupt is generated.
The TSnCSMD flag of TSnTMD register can select whether channel cyclic detection is executed a round or it is executed continuously.

15.3.2 Clock Source Selection

To execute the detection operation by Touch Sensor Timer, be sure to select the clock source of base counter by TSnCKMD register. The base counter consists of 10-bit counter.
The clock source cycle and the base counter cycle are determined by selecting the clock source.
Each task of the clock source cycle and the base counter cycle differs depending on the detection mode.

■ RC mode

In this mode, the clock source cycle determines the count cycle of base counter and the measurement resolution for detecting the period until the polarity of touch sensor input channel changes.
The base counter cycle is the maximum time which can execute the detection operation. When the input level of touch sensor input pin does not change within a cycle of the base counter, Touch detect error interrupt is generated and the detection operation in next channel starts.

■ OSC mode

The clock source cycle does not influence the detection operation.
The base counter cycle is the detection cycle of each channel.
The time required for the cyclic detection of all the channel can be calculated with "number of selected channels × detection cycle".

Table:15.3.1 Clock Source and Base Counter Cycle

TSnCKMD		Clock Source Cycle	Base Counter Cycle
TSnCK2 to TSnCK0			
000	fpll-div	50 ns	51.2 µs
001	fpll-div/2	100 ns	102.4 µs
010	fpll-div/4	200 ns	204.8 µs
011	fpll-div/8	400 ns	409.6 µs
100	fpll-div/16	800 ns	819.2 µs
101	fpll-div/32	1.6 µs	1.6 ms

* When fpll-div is 20 MHz

Set the TSnPSCEN flag to “1” after setting the TSnCK2 to TSnCK0 flags.

To change the value of TSnCK2 to TSnCK0 flags, be sure to set the TSnPSCEN flag to “0” in advance.

15.3.3 Touch Sensor Detection Resistor Selection

The composition of the resistor added to the touch sensor input pin can be selected either internal pull-up/pull-down resistor or external resistor by the TSnRSMD and TSnRCMD flags of TSnTMD register.
The composition of each resistor differs depending on the detection mode.

■ RC mode

- When the setting condition is "TSnRSMD = 0" (internal resistor) and "TSnRCMD = 0" (charge):
  Each touch sensor input pin is added the pull-up resistor.

- When the setting condition is "TSnRSMD = 0" (internal resistor) and "TSnRCMD = 1" (discharge):
  Each touch sensor input pin is added the pull-down resistor.

- When the setting condition is "TSnRSMD = 1" (external resistor):
  Each touch sensor input pin is not added the resistor.

■ OSC mode

- When the setting condition is "TSnRSMD = 0" (internal resistor):
  The composition of each touch sensor input pin is shown below.

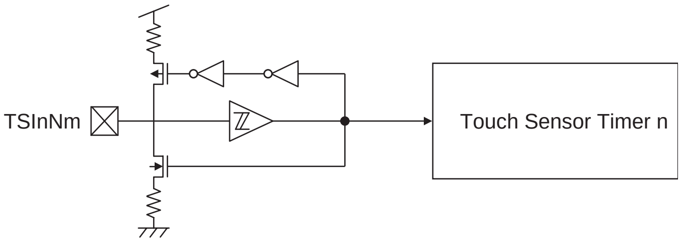


Figure:15.3.1 Composition of Touch Sensor Input pin in OSC mode

- When the setting condition is "TSnRSMD = 1" (external resistor):
  Pins TSnOP and TSnRC are used to connect the external resistor.
  Touch sensor input pin is connected to pin TSnRC with the analog switch inside LSI, when the corresponding channel is selected. The composition of each touch sensor input pin is shown below.

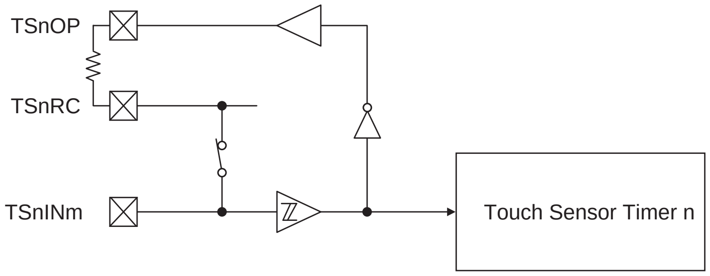


Figure:15.3.2 Composition of Touch Sensor Input pin in OSC mode

For pins TSnOP and TSnRC, the setup by the TSnMD and TSnRSMD flags of TSnTMD register is available regardless of the setup of port control registers.

15.3.4 RC Mode

In this mode, the base counter counts the period until the polarity of touch sensor input pin changes.
Charge mode ("Low" level to "High" level) or Discharge mode ("High" level to "Low" level) can be selected by TSnRCMD flag of TSnTMD register.

The detection operation in RC mode is shown below.
The touch sensor input pin is switched to input mode and the base counter starts counting.
When the polarity of touch sensor input pin changes, the data of the base counter as the measurement result is stored in TSnCHmCTDATH and TSnCHmCTDATL registers.

When the comparison result of the measurement result and the expected data satisfies the condition of "Expected data < Measurement result", touch detection interrupt is generated and the detection operation in next channel is started.

While touch sensor timer has halted, the value of TSnCHmCTDATH and TSnCHmCTDATL registers is loaded as the expected data.
When the value of TSnCHmEXDATH and TSnCHmEXDATL registers is updated while touch sensor timer is operating, the setup value of TSnCHmEXDATH and TSnCHmEXDATL registers is loaded as the next expected data after the detection operation of corresponding channel is completed.
When the input level of touch sensor input pin does not change within 1 cycle of the base counter, touch error interrupt is generated and the detection operation in next channel starts.

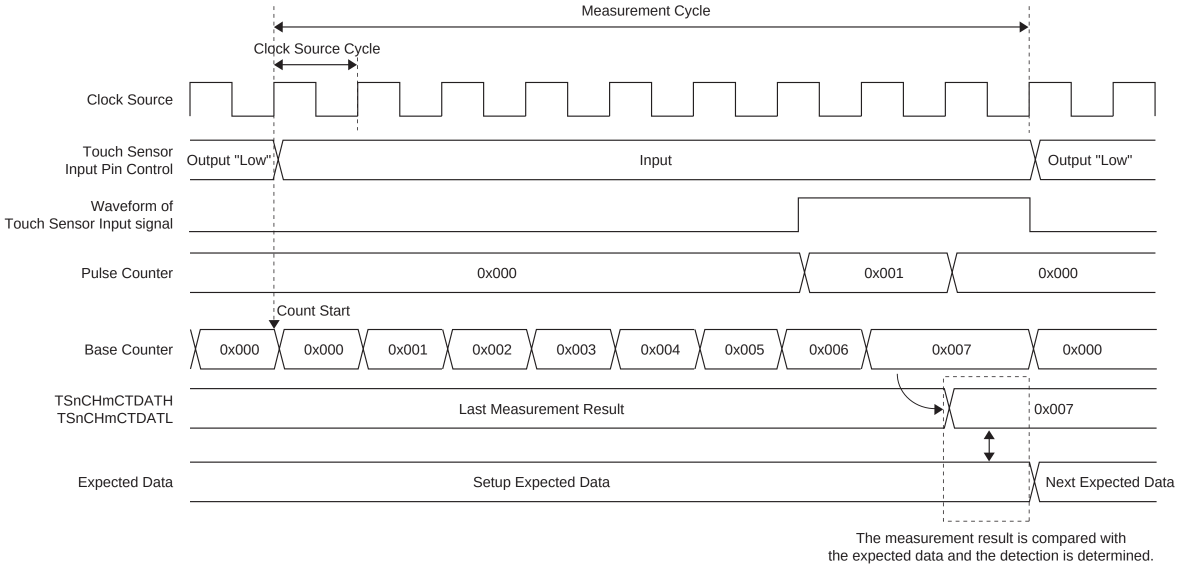


Figure:15.3.3 Detection Operation in RC Mode (Charge)

! When the detection operation is started in RC mode, the wait time for 1 cycle of base counter is inserted.

! When the touch sensor input channel is selected only one channel, the wait time for 1 cycle of base counter is inserted every detection operation.During the wait time, the pin state and the data of counter do not change.

! When Touch n detect error interrupt is occurred, the data of TSnCHmCTDATL and TSnCHmCTDATH registers is “0x3FF”.

15.3.5 OSC Mode

In this mode, the pulse counter counts the number of pulses input to touch sensor input pin during the cycle setup in base counter.

The detection operation in OSC mode is shown below.
The touch sensor input pin is switched to input mode and the base counter starts counting.
The data of the pulse counter equivalent to 1 cycle of base counter is stored in TSnCHmCTDATH and TSnCHm-CTDATL registers.

When the comparison result of the measurement result and the expected data satisfies the condition of "Expected data > Measurement result", touch detection interrupt is generated and the detection operation in next channel is started.

While touch sensor timer has halted, the value of TSnCHmEXDATH and TSnCHmEXDATL registers is loaded as the expected data.
When the value of TSnCHmEXDATH and TSnCHmEXDATL registers is updated while Touch Sensor Timer n is operating, the setup value of TSnCHmEXDATH and TSnCHmEXDATL registers is loaded as the next expected data after the detection operation of corresponding channel is completed.
When the data of pulse counter reaches "0x000" or the pulse counter overflows in 1 cycle of base counter, Touch n detect error interrupt is generated and the detection operation in next channel starts.

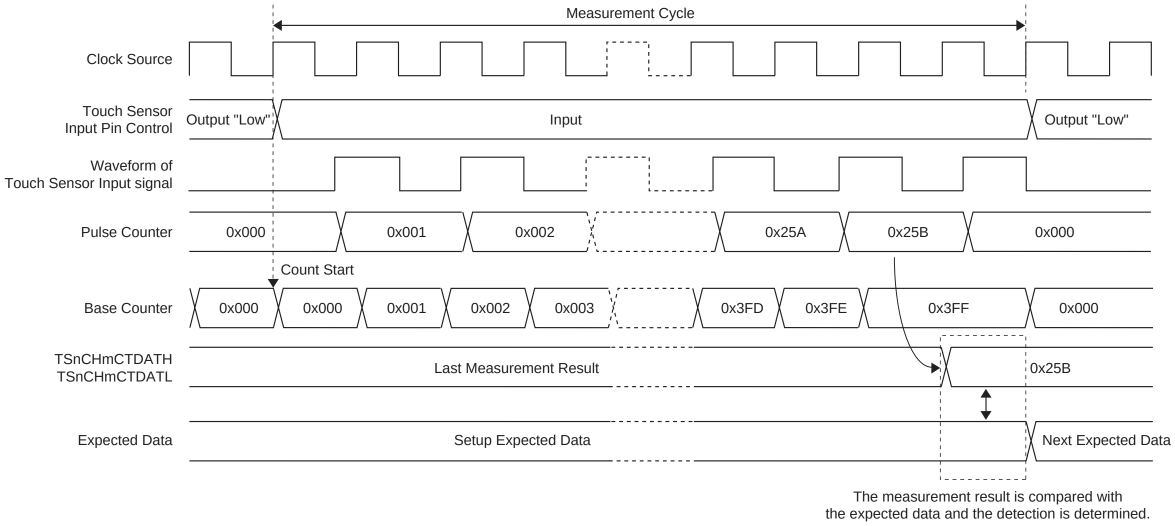


Figure:15.3.4 Detection Operation in OSC Mode

When the detection operation is started in OSC mode, the wait time for 1 cycle of base counter is inserted.

When the touch sensor input channel is selected only one channel, the wait time for 1 cycle of base counter is inserted every detection operation.During the wait time, the pin state and the data of counter do not change.

When touch n detect error interrupt is occurred, the data of TSnCHmCTDATL and TSnCHm-CTDATH registers is “0x3FF”.

15.3.6 Setup Example (Touch Sensor Timer)

■ Example of Setup for Touch Sensor Timer

Refer to the application note for detail.

15.4 Operation (Touch Sensor Data Automatic Transfer)

15.4.1 Overview of Data Automatic Transfer

The value of measurement result registers can be transferred to the arbitrary address in RAM by occurring each interrupt of Touch Sensor timer n as a trigger factor.

Data automatic transfer function is activated once, even if CPU is executing the instruction, microcontroller suspends normal operation in the timing which can release the bus. And it can transmit the data automatically by hardware using the bus released from CPU.

Data automatic transfer function is hardware different from interrupt function.
Therefore, setup of interrupt enable flag is not required, even if each interrupt of Touch Sensor timer n is selected as a trigger factor.

When interrupt selected as a trigger factor of data automatic transfer function and Touch n data transmission interrupt are used at same time, set each interrupt to different interrupt level according to interrupt priority.

15.4.2 Basic Operation and Timing

Basic operation and timing are described below.

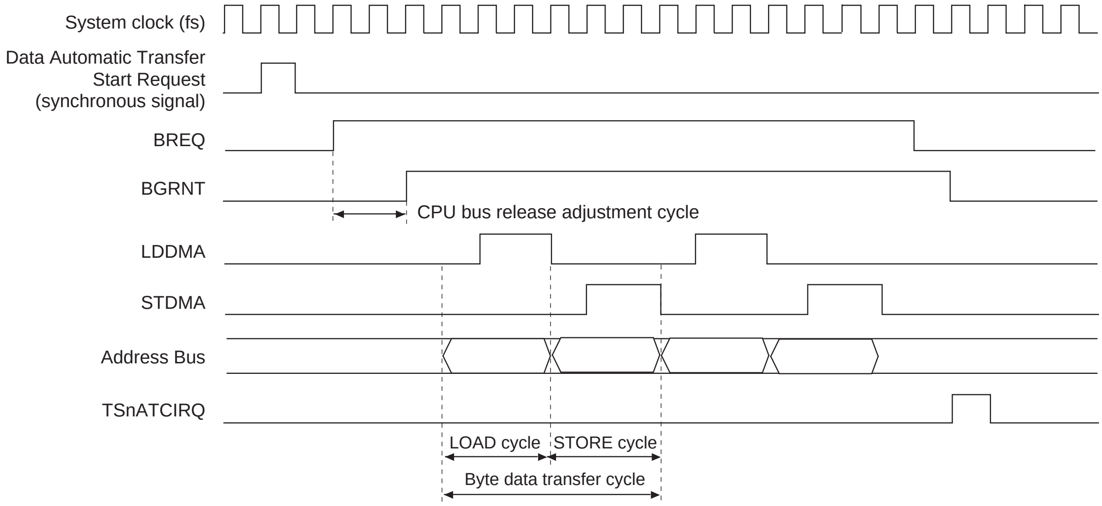


Figure:15.4.1 Data Automatic Transfer Operation Timing

■ Activation and Internal Bus Acquisition

Data automatic transfer function activates either when the selected interrupt factor occurs or when the activation flag (TSnATACT) is set by the software. Select the trigger factor in TSnATCNT1 register. When data automatic transfer function starts, data automatic transfer controller asserts bus release request signal (BREQ) to CPU.
When the CPU receives BREQ, it stops normal operation in the timing which can release the internal bus, even if it is executing the instruction. The period of minimum of 2 cycles is required from CPU receives BREQ until the internal bus is actually released.
After CPU releases the internal bus, it returns bus grant signal (BGRNT) to data automatic transfer controller. And microcontroller can transmit the data automatically by hardware using the bus released from CPU

When the software activation is selected as an trigger factor of data automatic transfer function, maximum of 4 instructions are required from setting of the TSnATACT flag until data automatic transfer function is activated.

■ Data Transfer

The basic transfer operation cycle is the "byte-data transfer cycle" in bytes of data. The "byte-data transfer cycle" consists of LOAD cycle and STORE cycle. In LOAD cycle, the data from the source register is read. In STORE cycle, the read data to the destination RAM address is stored. Repeat the required number of times for "byte-data transfer cycle" to transmit multi-byte of data continuously.

■ Transfer Completion

When all automatic transfer operation of setup data is completed, Touch n data transmission interrupt (TSnATIRQ) is generated and operation is stopped.

In both LOAD and STORE cycles, the operation of read and write accesses is the same as that of read and write accesses to RAM by the instruction of CPU.
Also, the wait settings for I/O space apply.
The following is the access timing for each memory space (when no-wait is set).

I/O space (special registers):	3 cycles
RAM space:	2 cycles

LOAD cycle and STORE cycle are set as follows;
Access time corresponding to each memory space + 1 cycle

In Figure:15.4.1, although the period from the rising of data automatic transfer start request signal to the starting of LOAD cycle depends on the state of CPU, it is required minimum of 6 cycles.

15.4.3 Setting of Data Automatic Transfer

■ Setting of Transfer Address to Memory Pointer

Set each start address of register space and RAM space for data automatic transfer in TSnATREGAP, TSnATRAMAPH and TSnATRAMAPL registers.
The lower 8 bits of register address can be set in TSnATREGAP register. The upper bits are fixed to "0x03D".
Among the total 10 bits of arbitrary RAM address, the upper 2 bits can be set in TSnATRAMAPH register, the lower 8 bits can be set in TSnATRAMAPL register

Each memory pointer can increment the address for every 1 byte of transmission using the equipped calculation function.
RAM address pointer can also disable to increment forcibly.

Set the transfer address while transfer operation is disabled (the TSnATEN flag of TSnATCNT0 register is "0").

■ Setting of Data Transfer Counter

Set the number of registers which is transferred in TSnATTRC register.
The data of TSnATTRC register is decremented for every 1 byte of transmission.When the data reaches "0x00", TSnATIRQ is generated and the transmission is completed.

Set the data transfer counter while transfer operation is disabled (the TSnATEN flag of TSnATCNT0 register is "0").

■ Setting of Trigger Factor and Control to Enable Transmission

Set the ATC trigger factor in TSnATCNT1 register.
Data automatic transfer is enabled by setting the TSnATEN of TSnATCNT0 register to "1"
To activate by software, in addition, set the TSnATACT of TSnATCNT0 register to "1".

When the data automatic transfer is completed, the data of TSnATREGAP, TSnATRAMAPH and TSnATRAMAPL registers is “the last transferred address + 1” and the data of TSnATTRC register is “0”.
To restart the data automatic transfer, set each register again.

15.4.4 Setup Example (Touch Sensor Data Automatic Transfer)

■ Example of Setup for Data Automatic Transfer

An example setup procedure is shown below.

Source Register Address: 0x03DC0 to 0x03DCF (TS0CH0CTDATL register to TS0CH7CTDATH)
Destination RAM Address: 0x003A0 to 0x003AF
Trigger Factor: Tstimer channel selection interrupt

Setup Procedure	Description
(1) Disable data automatic transfer function TS0ATCNT0(0x03E60) bp0: TS0ATEN = 0	(1) Set the TS0ATEN flag of TS0ATCNT0 register to “0” to disable data automatic transfer function.
(2) Set the register address pointer TS0ATREGAP(0x03E66) bp7 to 0: TS0ATREGAP7 to 0 = 0x40	(2) Set the lower 8bits of start address for source register in TS0ATREGAP register.
(3) Set the RAM address pointer TS0ATRAMAPL(0x03E63) bp7 to 0: TS0ATRAMAP7 to 0 = 0xA0 TS0ATRAMAPH(0x03E64) bp1 to 0: TS0ATRAMAP9 to 8 = 0x3	(3) Set the start address for destination RAM in TS0ATRAMAPH and TS0ATRAMAPL registers.
(4) Set the data transfer counter TS0ATTRC(0x03E62) bp7 to 0: TS0ATTRC7 to 0 = 0x10	(4) Set the number of registers which is transferred in TS0ATTRC register.
(5) Select the trigger factor TS0ATCNT1(0x03E61) bp1 to 0: TS0ATIR1 to 0 = 10	(5) Select tstimer channel selection interrupt as the trigger factor by the TS0ATIR1 to 0 flags of TS0ATCNT1 register.
(6) Enable data automatic transfer function TS0ATCNT0(0x03E60) bp0: TS0ATEN = 1	(6) Set the TS0ATEN flag of TS0ATCNT0 register to “1” to enable data automatic transfer function.

! To activate data automatic transfer function by the software, select the software interrupt as the trigger factor in step (5).
After the setup in step (1) to (6) is completed, set the TS0ATACT flag of TS0ATCNT0 register to “1”.
When the TS0ATACT flag is set, the data transfer is started. And then the TS0ATACT flag is cleared to “0” automatically.

Chapter 16 Internal Flash Memory

16.1 Overview

MN101EFAxG includes 128KB of a flash memory as an internal instruction memory.
MN101EFAxD includes 64KB of a flash memory as an internal instruction memory.
The flash memory is divided into the areas called "sector". In MN101EFAxG, there are seven sectors: 32 KB of Sector 0 to 2, 20 KB of Sector 3 and 4 KB of Sector 4 to 6. IN MN101EFAxD, there are five sectors: 32 KB of Sector 0, 20 KB of Sector 1 and 4 KB of Sector 2 to 4. This memory is writable in bytes and erasable in sectors.
For rewriting of the flash memory, it can be rewritten not only by a programmer but also by user program.
To rewrite the flash memory by user program, perform a subroutine call of command library stored in the dedicated area in advance from user program.

16.1.1 Outline Specification

Table:16.1.1 shows the outline specification of the flash memory.

Table:16.1.1 Outline Specification of Flash Memory

Function	Description
Capacity	128 KB (MN101EFAxG) 64 KB (MN101EFAxD)
Number of times of rewriting in each sector	32 Kbyte, 20 Kbyte Sector: 1000 times 4 Kbyte Sector: Over 10000 times
Voltage for rewriting	V_{DD5} : 4.0 V to 5.5 V
Voltage for reading	V_{DD5} : 4.0 V to 5.5 V
Data retention period	20 years
Flash memory rewriting method	Parallel programmer rewriting
	Serial programmer rewriting
	User mode microcontroller rewriting
	Boot mode microcontroller rewriting
Erasable unit	Sector
Programmable unit	Byte (Up to 64 byte of continuous writing is possible)
Data protection function	Protect function
	Security function

The number of times of rewriting is counted by the number of times of erasing sectors, and controlled on sector basis.
For example, if writing 1 byte of data in any sector in hundred times and then erasing the sector, a single rewriting is counted. Also, the number of times of rewriting in another sector, in which erasing is not performed, is not added.
Overwriting data is disabled. To rewrite data, write the data after erasing sectors.

16.1.2 Flash Memory Rewriting Method

This LSI uses two rewriting methods for writing and erasing in flash memory: Rewriting by a programmer (Programmer rewriting method) and rewriting by user program (Microcontroller rewriting method).
As a programmer rewriting method, parallel programmer rewriting method using parallel programmer and serial programmer writing method using serial programmer are provided.
As a microcontroller rewriting method, user mode microcontroller rewriting method is provided. In this method, the LSI is activated from user program and rewriting is performed during user program execution.

■ Parallel Programmer Rewriting Method

This method is generally used as a rewriting method of flash memory. Parallel programmer and a dedicated writer adapter are used to rewrite.
In this method, a single LSI before implemented on board can be rewritten.
By gang programmer, more than one LSI can be rewritten simultaneously. It is suitable for mass production of the LSI developed completely.

■ Serial Programmer Rewriting Method

This method is performed by using serial programmer.
In this method, the LSI implemented on board can be rewritten, and it is suitable for the uses of program development and parameter rewriting of individual equipment. Also, rewriting the LSI implemented on board allows more sufficient time to develop software than using parallel programmer rewriting method.

■ User Mode Microcontroller Rewriting Method

This method is performed by using serial programmer.
In this method, rewriting on any interface can be performed during user program execution after the LSI is implemented on board. It is suitable for program update during program execution and alternative EEPROM to retain data.

■ BOOT Mode Microcontroller Rewriting Method

This method is used to rewrite user program, and activated by the rewrite-only program written in the flash memory in advance.
In this method, rewriting on any interface can be performed when the LSI is implemented on board.
Because rewriting is performed by the updated without programmer even if the user program is destroyed.

Table:16.1.2 shows the outline of flash memory rewriting method.

Table:16.1.2 Outline of Flash Memory Rewriting Method

Rewriting method	Parallel programmer rewriting method	Serial programmer rewriting method	User mode microcontroller rewriting method	BOOT mode microcontroller rewriting method
Description	Rewriting by parallel programmer and rewriting adapter.	Rewriting by serial programmer	Rewriting after activation from user program execution.	Rewriting after activation from rewriting program execution.
Rewritable area	Sector 0 to 6 (MN101EFAxG) Sector 0 to 4 (MN101EFAxD)	Sector 0 to 6 (MN101EFAxG) Sector 0 to 4 (MN101EFAxD)	Sector 0 to 6 (MN101EFAxG) Sector 0 to 4 (MN101EFAxD)	Sector 0 to 6 (MN101EFAxG) Sector 0 to 4 (MN101EFAxD)
Required equipment	Parallel programmer	Serial programmer	-	-
	Rewriting adapter			
Features	Rewritable before board implementation	Rewritable after board implementation	Rewritable after board implementation	Rewritable after board implementation
	Rewriting program is not required.	Rewriting program such as serial communication control is not required	Programmer is not required	Programmer is not required.
	Multiple LSIs are rewritable simultaneously (When using gang programmer)	Prior writing is not required.	Rewritable in state of user program execution	Sector 6 (MN101EFAxG) or Sector 4 (MN101EFAxD) are used as activating area
	Prior writing is not required.	-	Rewriting program is required	Rewriting program is required

Table:16.1.3 and Table:16.1.4 show the memory map of the internal flash memory.

When the BOOT area is used, the memory map changes depending on the setting of the rewriting enable register (FBEWER).

Table:16.1.3 Memory Map of Internal Flash Memory (128 KB)

Sector	Sector Size	BOOT area is not used	BOOT area is used	
		MAIN area	MAIN area (FBEWER = 0x4b)	BOOT area (FBEWER ≠ 0x4b)
Sector 0	32 KB	0x04000 to 0x0BFFF	0x04000 to 0x0BFFF	(Accessing Prohibited)
Sector 1	32 KB	0x0C000 to 0x13FFF	0x0C000 to 0x13FFF	
Sector 2	32 KB	0x14000 to 0x1BFFF	0x14000 to 0x1BFFF	
Sector 3	20 KB	0x1C000 to 0x20FFF	0x1C000 to 0x20FFF	
Sector 4	4 KB	0x21000 to 0x21FFF	0x21000 to 0x21FFF	
Sector 5	4 KB	0x22000 to 0x22FFF	0x22000 to 0x22FFF	
Sector 6 (also used as BOOT area)	4 KB	0x23000 to 0x23FFF	0x23000 to 0x23FFF	0x04000 to 0x04FFF

Table:16.1.4 Memory Map of Internal Flash Memory (64 KB)

Sector	Sector size	BOOT area is not used	BOOT area is used	
		MAIN area	MAIN area (FBEWER = 0x4B)	BOOT area (FBEWER ≠ 0x4B)
Sector 0	32 KB	0x04000 to 0x0BFFF	0x04000 to 0x0BFFF	(Accessing Prohibited)
Sector 1	20 KB	0x0C000 to 0x10FFF	0x0C000 to 0x10FFF	
Sector 2	4 KB	0x11000 to 0x11FFF	0x11000 to 0x11FFF	
Sector 3	4 KB	0x12000 to 0x12FFF	0x12000 to 0x12FFF	
Sector 4 (also used as BOOT area)	4 KB	0x13000 to 0x13FFF	0x13000 to 0x13FFF	0x04000 to 0x04FFF

16.1.3 Flash Memory Area

Flash memory is divided into three areas: MAIN area, BOOT area, reserved area.
MAIN area is to store user program, and the area is rewritable.
BOOT area is to activate BOOT mode microcontroller rewriting method, and the area is rewritable.
Reserved area is to store command library to use for flash memory rewriting, and the area is not rewritable. The capacity of the reserved area is not included in the flash memory capacity.

16.1.4 Protect Function

Protect function can disable writing or erasing to prevent write errors or erase errors for the flash memory in which data has been written. Protection can be set in sectors. Once protection is set, it cannot be released.

! Note that once protection is set, writing and erasing cannot be performed.

16.1.5 Security Function

Security function prevents alteration or leakage of the program using programmer or debugger.
Writing, reading, and erasing for all the area of the flash memory can be disabled by security function. To set or release the security, a key code (128 bit) is used. Once the key code is set, it cannot be changed.

! When enabling the security function, collectively erasing cannot be executed. Note that writing or erasing cannot be performed if the key code is lost.

16.2 Parallel Programmer Rewriting

Parallel programmer rewriting is used to rewrite the flash memory with a rewriting adapter and parallel programmer corresponding to this LSI. Rewriting of the flash memory is possible for the LSI before implemented board. For parallel programmer writing, All sectors can be rewritten.

16.2.1 Equipment Configuration in Parallel Programmer Rewriting

Perform rewriting by attaching a rewriting adapter dedicated for this LSI to the parallel programmer corresponding to this LSI. For the parallel programmer, the following two programmers are provided: programmer which can rewrite without host computer, and programmer for gang programming to rewrite multiple LSIs collectively. For procedure of rewriting, refer to the parallel programmer instruction manual.

Our website [Semiconductor Technical Support System] shows the most up-to-date information on rewriting adapter and parallel programmer. For our website URL, refer to the end of this manual.

16.3 Serial Programmer Rewriting

Serial programmer rewriting is used to rewrite the flash memory with the serial programmer corresponding to this LSI. The flash memory can be rewritten during implementation of the LSI on the circuit board.
It is not required to write serial communication control program in the flash memory in advance.
For serial programmer writing, All sectors can be rewritten.

16.3.1 Equipment Configuration in Serial Programmer Rewriting

Install a connector to connect the serial programmer to this LSI on the circuit board. Connect the serial programmer via this connector. For the serial programmer, the programmer which can rewrite without host computer is provided.
For procedure of rewriting, refer to the serial programmer instruction manual.

Our website [Semiconductor Technical Support System] shows the most up-to-date information on rewriting adapter and serial programmer. For our website URL, refer to the end of this manual.

16.3.2 Pin Configuration in Serial Programmer Rewriting

Table:16.3.1 shows the pin and signal functions in serial programmer rewriting, and Figure:16.3.1 shows the pin connecting diagram.

Table:16.3.1 Pin and Signal Functions

Pin	I/O	Function	Remarks
VDD5	-	Power supply	-
VSS	-	GND	-
NRST	I	Reset pin	-
P00/OCD_DATA	I/O	Data I/O pin	-
P01/OCD_CLK	I	Clock input pin 1	-
DMOD	I	Mode setting pin	-
MMOD	I	Activation mode setting pin	Fix to “0” in serial programmer rewriting
VREF+	-	Reference voltage	Whether using the A/D function, the voltage difference between V_{REF+} and V_{SS} must be set to more than V_{DD5}. Set the condition of V_{DD5}=V_{REF+}.

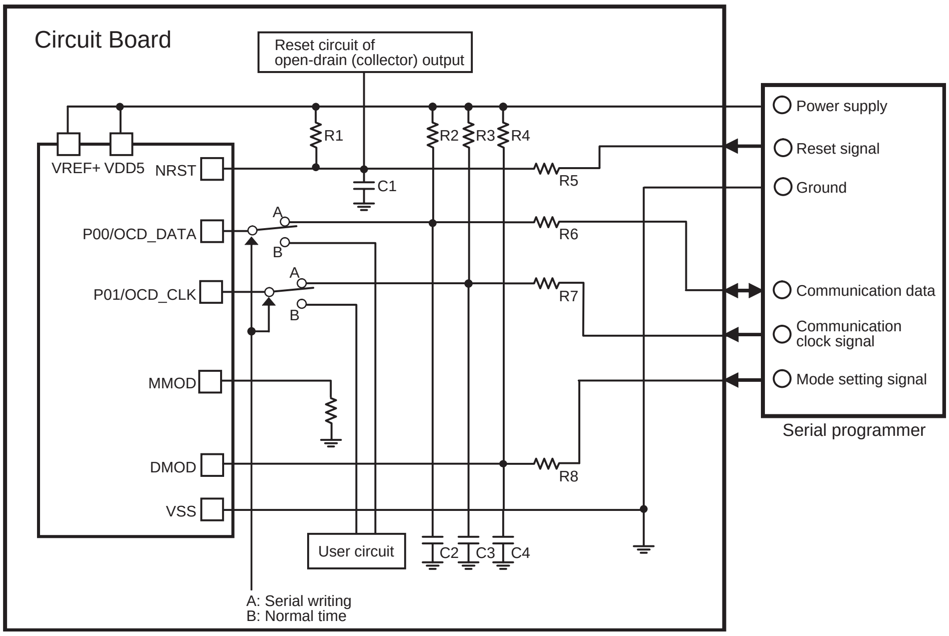


Figure:16.3.1 Pin Connecting Diagram in Serial Programmer Rewriting *1

*1: The pin connecting diagram is an example. The pin to control and the external circuit differ depending on programmer.
Refer to the manual of the programmer for detail.

16.4 Component Value Calculations

This section presents the calculations for each of the components used for connection.

16.4.1 Component Values

The values of the components used in Figure:16.3.1 are shown below.

- The value of Resistor R1 must be greater than R_{upRst}.
- The values of Resistors R2, R3 and R4 must be greater than R_{upMin} and greater than 1 kΩ.
- The value of Resistor R5 must be less than 1/10 that of R1 and less than R_{sMax}.
- The value of Resistor R6 must be less than 1/10 that of R2 and less than R_{sMax}.
- The value of Resistor R7 must be less than 1/10 that of R3 and less than R_{sMax}.
- The value of Resistor R8 must be less than 1/10 that of R4 and less than R_{sMax}.
- C1 must be less than C_{rst} and less than 100 µF.
- C2, C3 and C4 must be under 50 pF.
- Except for the VDD5 and VSS lines, the line length of the signal from the connector to the microcomputer must be less than 50 cm.

This document is written assuming that the clock and data outputs used from programmer are the push-pull outputs. The required component values will differ from those shown here if programmer open-drain outputs are used.

16.4.2 Reset Signal Capacitor (C1) Maximum Value Calculation

After applying a reset to the target microcomputer, programmer waits for a fixed period and then initiates communication. Determine the maximum value, C_{rst}, of the reset signal capacitor from Formula (1) so that the sum of the time constant and the oscillator stabilization time (Twait) is under 250 ms.

$$C_{rst} = \frac{250 \times 10^{-3} - T_{wait}}{R1} \qquad \text{.... Formula (1)}$$

16.4.3 Pull-up Resistor (R1) Minimum Value Calculation

If the maximum output current from programmer is 12 mA. Since this value is the maximum load current available for outputting a low level, R_{upRst} can be determined from Formula (2).

$$R_{upRrst} = \frac{\text{Operating supply voltage } (V_{DD5})}{12\ \text{mA}} - R_{sRrst} \qquad \text{.... Formula (2)}$$

16.4.4 Relationship Between R_{upRst} and R_{sRt}

If a resistor with a large value is inserted in series with the reset pin, the pull-up resistor (R_{upRst}) and the series resistor (R_{sRst}) must meet the condition in Formula (3) to ensure that the signal level falls all the way to the low level.

$$R_{upRrst} \times \frac{1}{10} \geq R_{sRrst} \qquad \text{.... Formula (3)}$$

16.4.5 Pull-up Resistor (R2, R3 and R4) Minimum Value Calculations

Find the maximum output current, I_{OL}, for the pins used for communication from [Chapter 1 1.5.3 DC Characteristics]. Since that value is the maximum load current available to output low level, R_{upMin} can be determined according to Formula (4).

$$R_{upMin} = \frac{\text{Operating supply voltage } (V_{DD5})}{\text{Pin maximum output current } (I_{OL})} \qquad \text{.... Formula (4)}$$

16.4.6 Communication Pin Series Resistor (R6, R7, R8) Maximum Value Calculations

If series resistors are inserted in the communication pin lines, the signal transmission speed will be slowed due to the influence of the load capacitors (C2, C3). To assure reliable communication, the time for the signal voltage to change by 63 % of the supply voltage (i.e. the time constant) must be held to under 1/8 of the communication period.
When load capacity is 50 pF and the communication frequency is 1 MHz, the maximum resistor value (R_{sMax}) for stable communication will be 2.5 kΩ by calculations according to Formula (5).

$$R_{sMax} = \frac{1}{8 \times \text{Communication frequency (f)} \times \text{Load capasitance (C)}} \quad \text{.... Formula (5)}$$

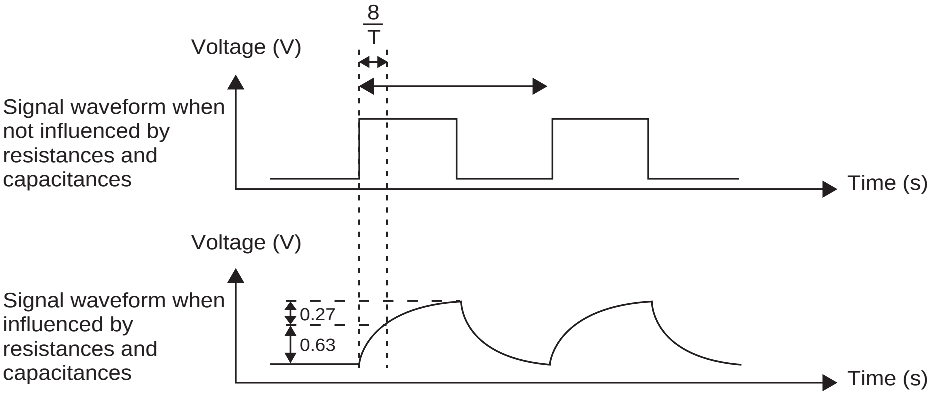

8
T
Voltage (V)
Signal waveform when not influenced by resistances and capacitances
Time (s)
Voltage (V)
Signal waveform when influenced by resistances and capacitances
0.27
0.63
Time (s)

Figure:16.4.1 Relationship Between Communication Frequency, R and C

When using on-board debugger, set the resistance value of 100 Ω or lower, and connect it in series with a communication terminal.

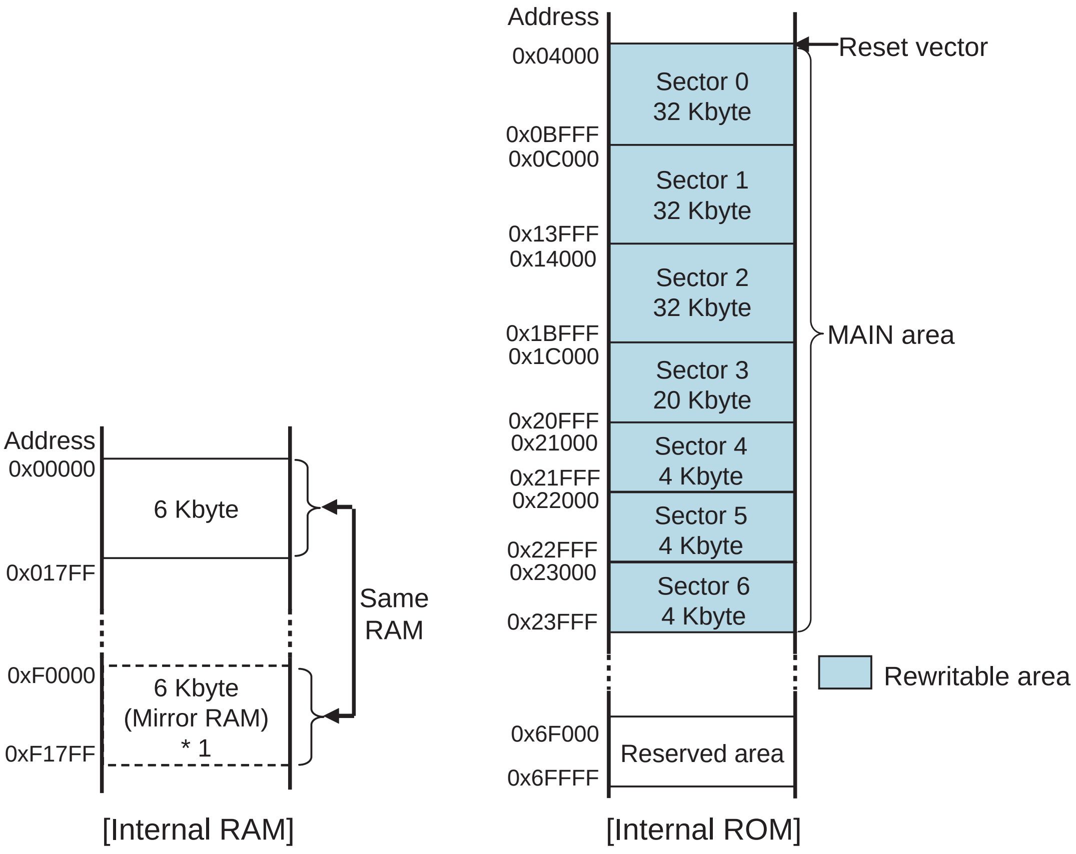
16.4.7 Relationship Between R_{upMin} and R_{sMax}

It is possible to insert resistors with large values in series with the communication lines if the communication speed is lowered. However, even in that case, the pull-up resistor (R_{upMin}) and the series resistor (R_{sMax}) must meet the condition in Formula (6) to ensure that the signals fall to the low level.

$$R_{upMin} \times \frac{1}{10} \geq R_{sMax} \quad \text{.... Formula (6)}$$

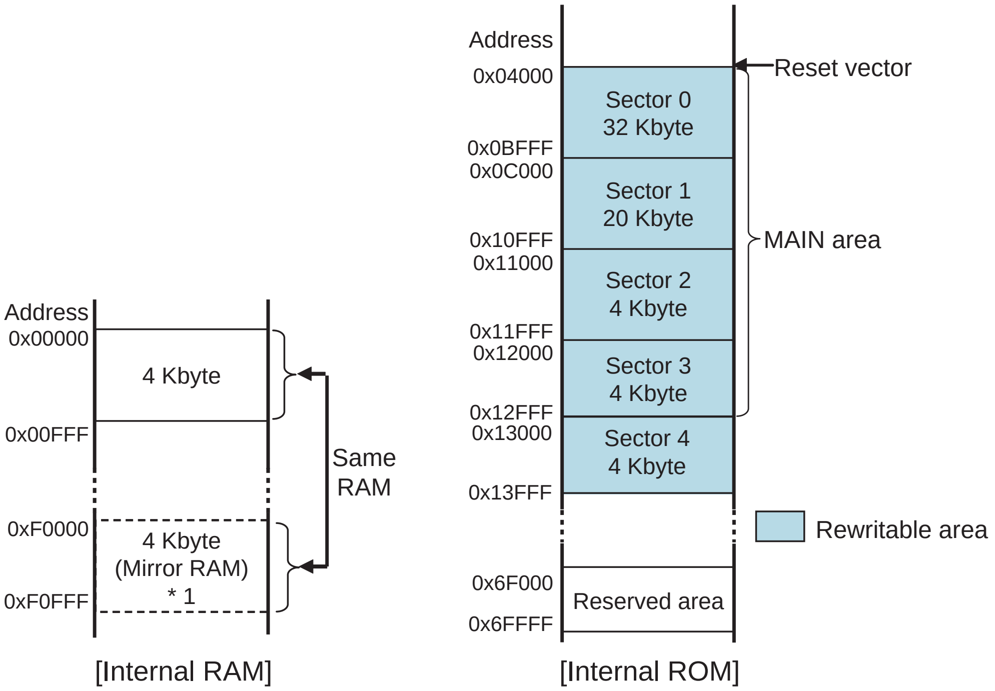
16.5 User Mode Microcontroller Rewriting

User mode microcontroller rewriting is used to rewrite the internal flash memory by the rewriting program included in the user program in advance. On any interface, rewriting during user program execution after implemented on board is possible.
It is suitable for a program which update during user program execution, or for alteration of EEPROM which retains data.
In this rewriting method, the entire area can be rewritten.
To set the user mode microcontroller rewriting method, release reset during the activation mode setting pin MMOD = "Low".
Figure:16.5.1 and Figure:16.5.2 show the memory map in the user mode microcontroller rewriting.



*1: Mirror RAM: Refer to [Chapter 2 2.2 Memory Space].

Figure:16.5.1 Memory Map in User Mode Microcontroller Rewriting(128 KB)



*1: Mirror RAM: Refer to [Chapter 2 2.2 Memory Space].

Figure:16.5.2 Memory Map in User Mode Microcontroller Rewriting(64 KB)

When reset is released while the activation mode setting pin MMOD = "Low", Sector 0 to 6 [Sector 0 to 4] are addressed in 0x04000 to 0x23FFF[0x13FFF], and activated from 0x04000. When 0x4B is set in the rewriting enable register (FBEWER), reserved area is addressed in 0x6F000 to enable command library, and Sector 0 to 6 [Sector 0 to 4] can be rewritten.
Flash memory rewriting is performed from user program. Therefore, when the program is accidentally destroyed, the LSI could no longer execute a proper operation. At that time, use the serial programmer rewriting method to rewrite the program correctly.

16.5.1 Rewriting by Command Library

The flash memory can be easily rewritten by a subroutine call of command library inside the program. The command library is stored in the reserved area of the flash memory.
If the subroutine-call is executed to the command library, the program required for rewriting is automatically transferred from the reserved area of the flash memory to the internal RAM. If the rewriting of the flash memory is controlled by the program on the internal RAM after the program is transferred and the rewriting is completed, it returns from the command library. The subroutine call of the command library is executed from the program on the flash memory. But if the sector, where the program to call the command library exists, is erased, the returning address from the command library is erased and the microcontroller does not operate.When rewriting the sector where the rewriting program exists, the rewriting program needs to be allocated in the internal RAM and the command library is called up from the internal RAM.

Refer to [Chapter 16 16.5.3 Rewriting Procedure-2] for details.

16.5.2 Rewriting Procedure-1

Figure:16.5.3 shows a sample of the rewriting procedure suitable for rewriting a part of the flash memory. This procedure is not used for the rewriting of the sector where the rewriting program stores. Please refer to [Chapter 16 16.5.3 Rewriting Procedure-2] when rewriting the sector including the rewriting program.

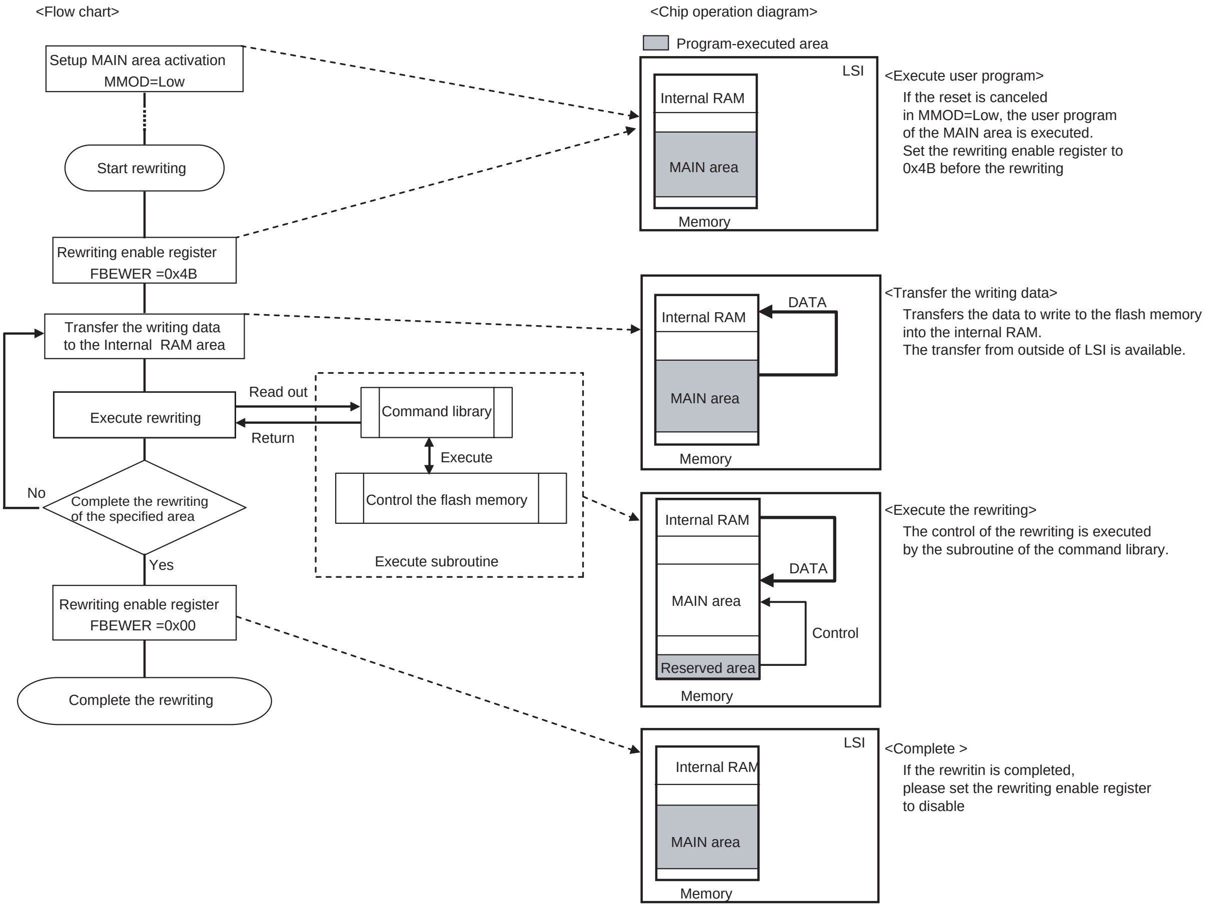
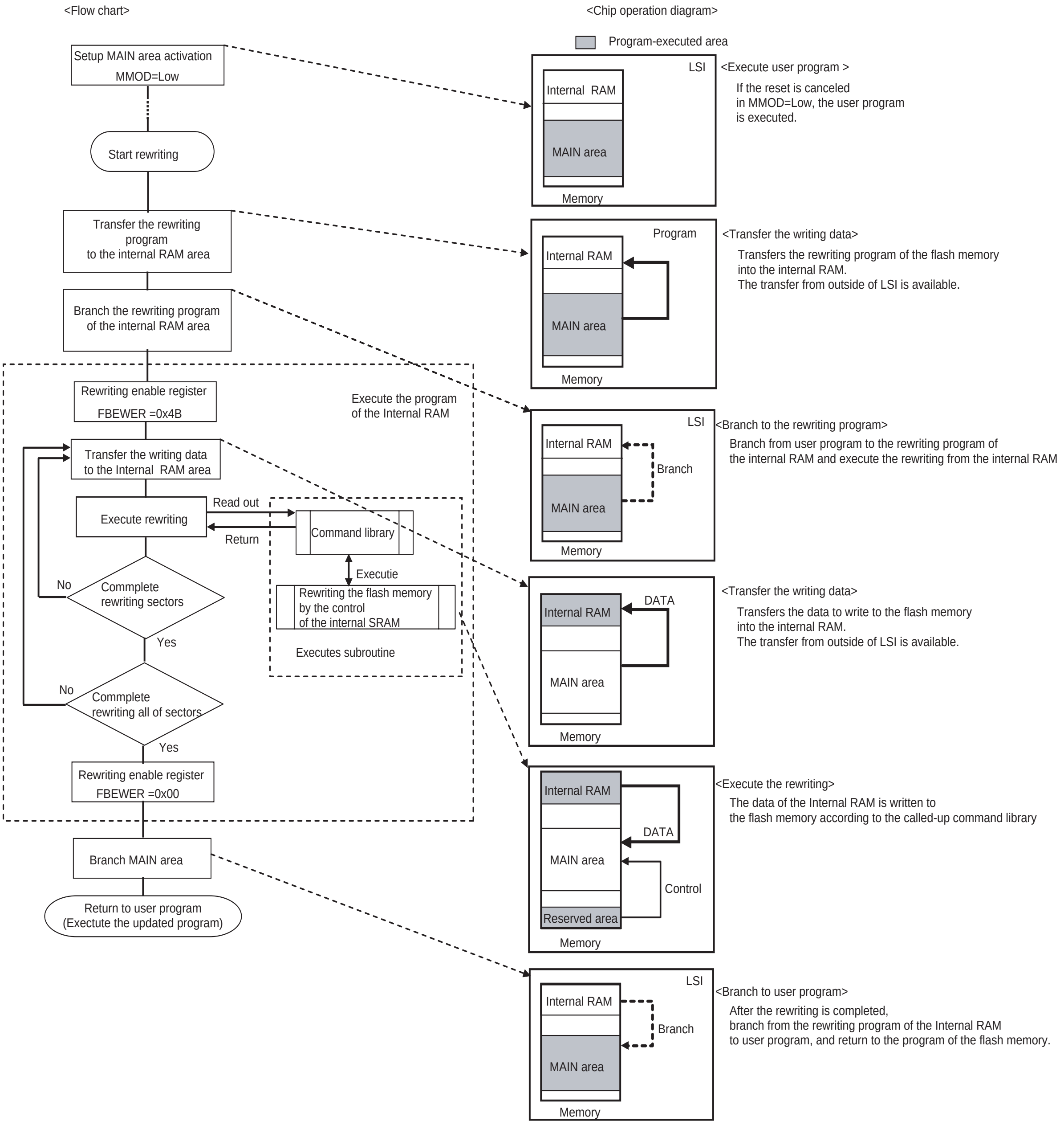


Figure:16.5.3 Rewriting Procedure-1 in User Mode Microcontroller Rewriting

16.5.3 Rewriting Procedure-2

In order to update the sector including the rewriting program, the rewriting program must be executed by the internal RAM. Figure:16.5.4 shows a sample procedure to execute the rewriting program by the internal RAM to rewrite the flash memory.
The following procedure shows a sample of whole flash memory rewriting, but it is possible to specify a certain sectors and rewrite them.


<Flow chart>
<Chip operation diagram>
Program-executed area
Setup MAIN area activation
MMOD=Low
Start rewriting
Transfer the rewriting program to the internal RAM area
Branch the rewriting program of the internal RAM area
Rewriting enable register
FBEWER =0x4B
Execute the program of the Internal RAM
Transfer the writing data to the Internal RAM area
Execute rewriting
Read out
Return
Command library
Executie
Rewriting the flash memory by the control of the internal SRAM
Executes subroutine
No
Commplete rewriting sectors
Yes
No
Commplete rewriting all of sectors
Yes
Rewriting enable register
FBEWER =0x00
Branch MAIN area
Return to user program (Exectute the updated program)
LSI
Internal RAM
MAIN area
Memory
Program
Branch
DATA
Control
Reserved area

<Execute user program >
If the reset is canceled in MMOD=Low, the user program is executed.

<Transfer the writing data>
Transfers the rewriting program of the flash memory into the internal RAM.
The transfer from outside of LSI is available.

<Branch to the rewriting program>
Branch from user program to the rewriting program of the internal RAM and execute the rewriting from the internal RAM

<Transfer the writing data>
Transfers the data to write to the flash memory into the internal RAM.
The transfer from outside of LSI is available.

<Execute the rewriting>
The data of the Internal RAM is written to the flash memory according to the called-up command library

<Branch to user program>
After the rewriting is completed, branch from the rewriting program of the Internal RAM to user program, and return to the program of the flash memory.

Figure:16.5.4 Rewriting Procedure-2 in User Mode Microcontroller Rewriting

16.6 BOOT Mode Microcontroller Rewriting

BOOT mode microcontroller rewriting is used to rewriting the internal flash memory by the rewrite-only program written in BOOT area in advance. In this rewriting method, rewriting is performed by the rewrite-only program independent of user program. As a result, user program can be updated without programmer even if the user program is destroyed by mistake.
In this rewriting method, the entire area can be rewritten. However, if BOOT area is rewritten by mistake, rewriting with this method cannot be performed.
The LSI activates by releasing reset during the activation mode setting pin MMOD = "High".

16.6.1 Rewriting Procedure

It is necessary to write the program for rewriting in advance by using serial programmer rewriting method or pararell programmer rewriting method.

Figure:16.6.1 and Figure:16.6.2 show the memory map in BOOT mode microcontroller rewriting.

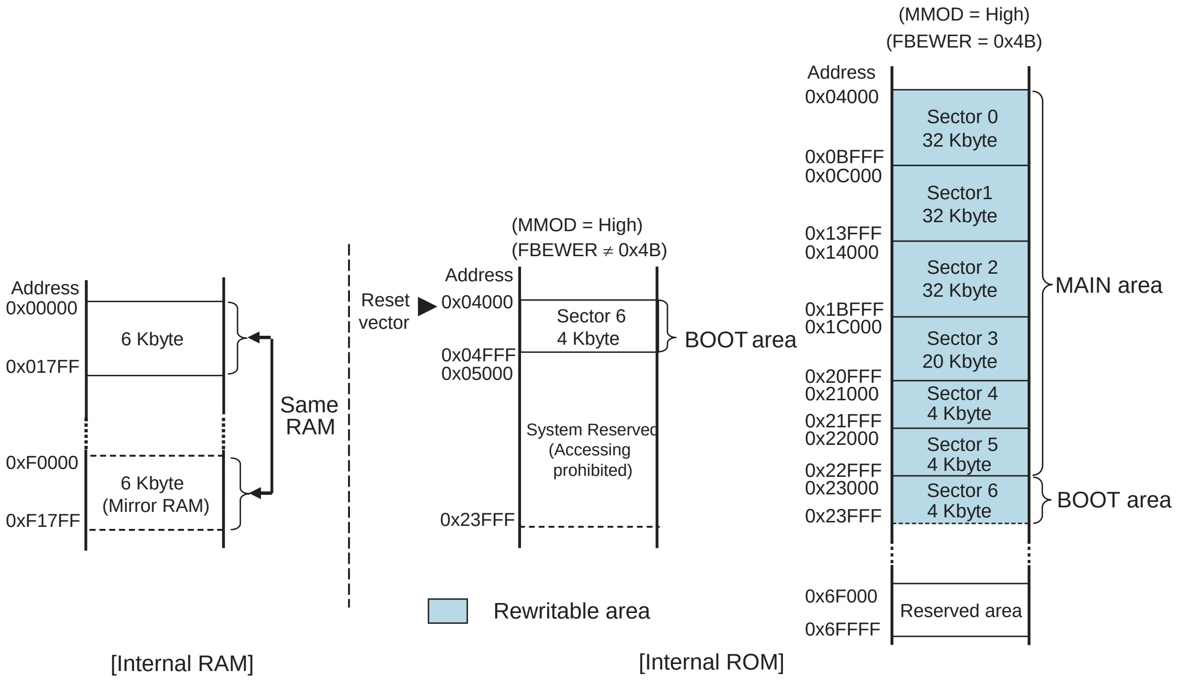


Figure:16.6.1 Memory Map in BOOT Mode Microcontroller Rewriting (128 KB)

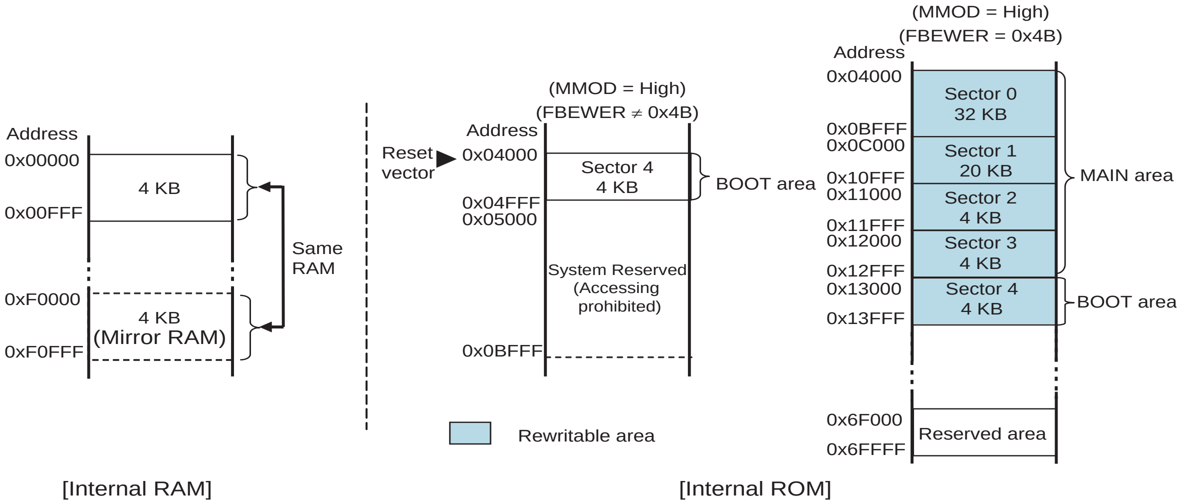


Figure:16.6.2 Memory Map in BOOT Mode Microcontroller Rewriting (64 KB)

When releasing reset during the activation mode setting pin MMOD = "High", BOOT (Sector 6 [Sector 4]) area is addressed to "0x04000" and the LSI activates from BOOT area.
If the rewriting enable register (FBEWER) is set to "0x4B", the memory map is changed. MAIN area (Sector 0 to 5 [Sector 0 to 3]) ia addressed to "0x04000" and BOOT area (Sector 6 [Sector 4]) is addressed "0x23000 [0x13000]". In addition, the reserved area is addressed to "0x6F000" and the command library becomes available to enable the rewriting of MAIN area (Sector 0 to 5 [Sector 0 to 3]) and BOOT area (Sector 6 [Sector 4]).
If the program is executed from BOOT area when the memory map is changed, programming and reading data will be disabled. The setting routine of the rewriting enable register must be allocated in the internal RAM and executed in the internal RAM.

After rewriting, the activation mode setting pin MMOD is set to "Low" to execute reset, the rewritten user program which is activated from MAIN area will be executed.
Figure:16.6.3 shows a sample procedure of rewriting by BOOT activation.

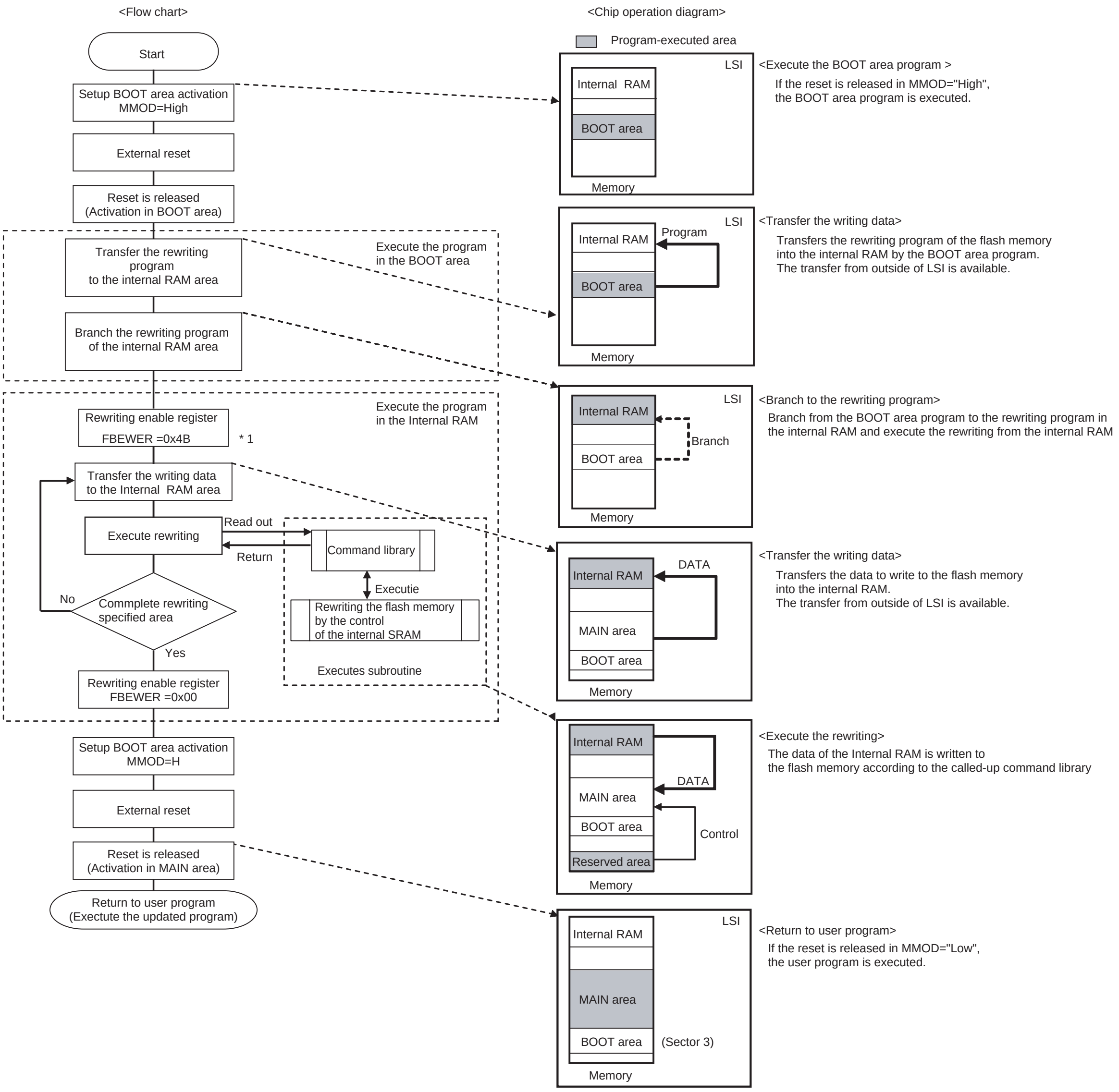

<Flow chart>
Start
Setup BOOT area activation MMOD=High
External reset
Reset is released (Activation in BOOT area)
Transfer the rewriting program to the internal RAM area
Branch the rewriting program of the internal RAM area
Execute the program in the BOOT area
Rewriting enable register FBEWER =0x4B
* 1
Execute the program in the Internal RAM
Transfer the writing data to the Internal RAM area
Execute rewriting
Read out
Return
Command library
Executie
Rewriting the flash memory by the control of the internal SRAM
Executes subroutine
No
Commplete rewriting specified area
Yes
Rewriting enable register FBEWER =0x00
Setup BOOT area activation MMOD=H
External reset
Reset is released (Activation in MAIN area)
Return to user program (Exectute the updated program)
<Chip operation diagram>
Program-executed area
LSI
Internal RAM
BOOT area
Memory
<Execute the BOOT area program >
If the reset is released in MMOD="High", the BOOT area program is executed.
Program
<Transfer the writing data>
Transfers the rewriting program of the flash memory into the internal RAM by the BOOT area program. The transfer from outside of LSI is available.
Branch
<Branch to the rewriting program>
Branch from the BOOT area program to the rewriting program in the internal RAM and execute the rewriting from the internal RAM
DATA
MAIN area
<Transfer the writing data>
Transfers the data to write to the flash memory into the internal RAM. The transfer from outside of LSI is available.
Control
Reserved area
<Execute the rewriting>
The data of the Internal RAM is written to the flash memory according to the called-up command library
(Sector 3)
<Return to user program>
If the reset is released in MMOD="Low", the user program is executed.

Figure:16.6.3 Rewriting Procedure in BOOT Mode Microcontroller Rewriting

*1 If the rewriting enable register (FBEWER) is set to "0x4B" in BOOT area, the memory map is changed to disable programming and reading of data. The setting routine of rewriting enable register must be allocated in the internal RAM and executed on the internal RAM.
(In user mode microcontroller rewriting method, this constraint is not available.)

16.6.2 Protection of BOOT Area

If BOOT area is rewritten by mistake, rewriting with BOOT mode microcontroller rewriting method will not be performed. To use Sector 6 [Sector 4] as BOOT area, it is recommended to protect Sector 6 [Sector 4] in advance.

16.7 Appendix

16.7.1 Rewriting Control Register

Table:16.7.1 shows the rewriting control register.

Table:16.7.1 Rewriting Control Register

Rewriting Control Register		
Registers	FBEWER	
Address	0x03FBD	
R/W	R/W	
Initial value	0x00	
Function	Setup writing/erase enable of the flash memory	
Setup	FBEWER=0x4B	Enable writing/erase enable of the flash memory Command library is available.
	FBEWER≠0x4B	Disable writing/erase enable of the flash memory When using BOOT mode microcontroller rewriting method, BOOT area is addressed at 0x04000.

16.7.2 Mode Setting

Table:16.7.2 shows the pin setup in microcontroller rewriting mode.
When the reset is released, the status of MMOD pin determines the microcontroller rewriting mode. Microcontroller rewriting mode can be activated from MAIN area or BOOT area.

Table:16.7.2 Pin Setup in Microcontroller Rewriting Mode

Activation Area	Rewriting Area	Pin Setup	
		NRST	MMOD
MAIN area	MAIN area	Low → High	Low
BOOT area	MAIN area + BOOT area		High

16.7.3 Procedure

■ Whole Chip Rewriting Procedure

Figure:16.7.1 shows whole chip rewriting procedure.
First of all, erase all of sectors by the control command. The erase is executed by the sector unit and multiple sectors can be specified.
Next, execute writing by the control command. At that time, up to 64-Byte consecutive writing is possible. But writing crossing over the sectors is not possible.

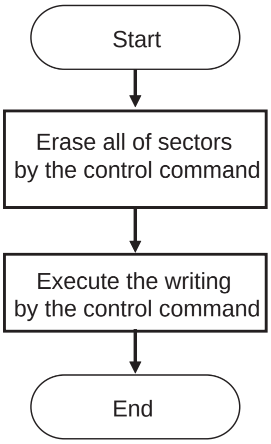

Start
Erase all of sectors by the control command
Execute the writing by the control command
End

Figure:16.7.1 Whole Chip Rewriting Procedure

■ Sector Unit Rewriting Procedure

Figure:16.7.2 shows rewriting procedures by sector.
First, erase sectors to be rewritten using the control command.
Next, execute writing by a control command.
To rewrite multiple sectors, multiple sectors can be specified by a control command. Writing crossing over sectors cannot be executed.

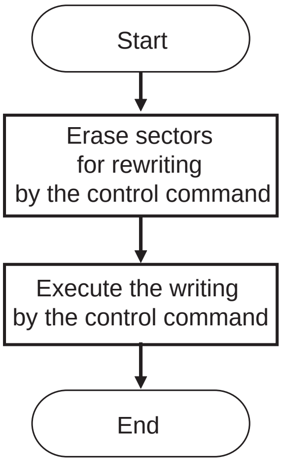

Start
Erase sectors for rewriting by the control command
Execute the writing by the control command
End

Figure:16.7.2 Sector Unit Rewriting Procedure

■ Partial Sectors Rewriting Procedure

Figure:16.7.3 shows partial sector rewriting procedure.
The erase can be executed only by the sector unit. If rewriting partial sector, back-up all the data in the sector to be rewritten before the erase, and execute the writing of the revised data after the erase. But if all the data in the sectors to be rewritten can be transferred from an external device, the data back-up is not necessary. In this case, please erase the target sectors and transfer the data from the external device for writing.

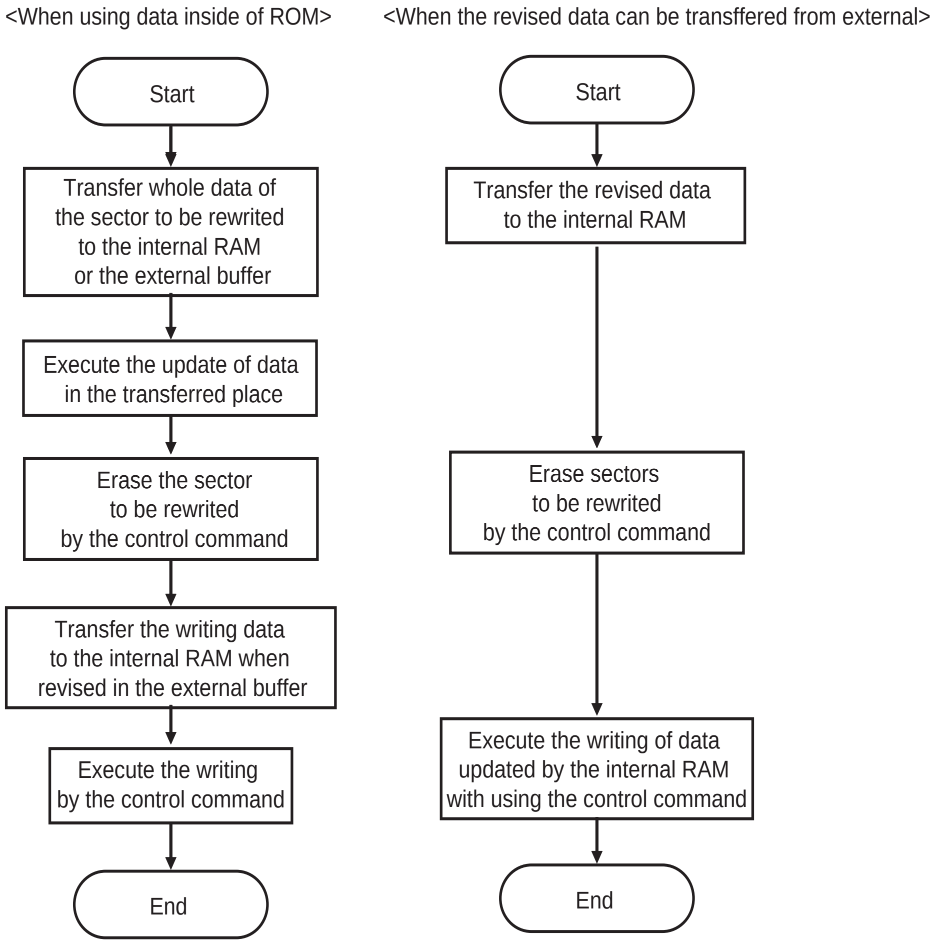

<When using data inside of ROM>
Start
Transfer whole data of the sector to be rewrited to the internal RAM or the external buffer
Execute the update of data in the transferred place
Erase the sector to be rewrited by the control command
Transfer the writing data to the internal RAM when revised in the external buffer
Execute the writing by the control command
End
<When the revised data can be transffered from external>
Start
Transfer the revised data to the internal RAM
Erase sectors to be rewrited by the control command
Execute the writing of data updated by the internal RAM with using the control command
End

Figure:16.7.3 Partial Sectors Rewriting Procedure

16.7.4 Precautions

The flash memory rewriting is executed by the subroutine call at the initial address of the command library by using the JSR instruction. In rewriting, please pay attention to the following.

- If the command library is executed by the subroutine call, the rewriting enable register (FBEWER) must be set to "0x4B". When the rewriting is completed, other value except "0x4B" must be set to the rewriting enable register (FBEWER) in order to avoid error in writing.
  And the subroutine call of the command library is prohibited when the rewriting enable register (FBEWER) is not set to "0x4B". As the reserved area where the command library is stored is not allocated, a non-maskable interrupt is generate by a undefined instruction.

- Please execute subroutine-call to the command library, after the initial address of the structure which stores the rewriting parameter is stored to address register A0. The executed result of the command library is stored in data register D0.
  The command library is a common specification that the argument is stored in address register A0 and the return value is stored in data register D0, and can be called up from C program.
  Refer to [Chapter 16 16.7.5 Command Library] and the attached papers for the overview of the command library and for specification, respectively.

- The command library needs to the work area and stack area except the structure which stores the rewriting parameter. The work area can be set by ram_adr parameter. The structure and work area must be on the internal RAM. The required work area is 260 byte for the Sector_Sum_Verify_Lib. The other command library needs up to 64 byte as the work area. The stack area needs up to 47 byte.

- A part of the command library can accept interrupts during command library execution. During command library execution, the interrupt vector base is automatically changed to "0x00100". Refer to the attached paper for the interrupts during command library execution.

- To execute rewriting flash memory using command library, it is necessary to set watchdog error detect cycles for watchdog timers as indicated below.

  Watchdog timer: Watchdog error detect cycle = "2^{20} × system clock cycle"

- After the CPU operation mode is set to NORMAL mode, call the command library.

- The maximum work area at using command library is different for partial engineering sample. When use the engineering sample, consult our offices.

16.7.5 Command Library

Following are the functions as command library for flash memory rewriting.
The flash memory rewriting is executed by the subroutine call at the initial address of the command library by using the JSR instruction.

Table:16.7.3 Command library

Command library	Argument	Return value	Function
Program_Lib	ram_adr dst_adr dst_bank bc src_adr	Normal end Writing error 1over0 error Security error Protect error Address volation error Double activation error Double activation error High-speed interrupt time over error	Write bc+1 bytes of data stored in src_adr to dst_bank,dst_adr
Erase_Lib	ram_adr sector_number	Normal end Erase error Security error Protect error Address volation error Double activation error Double activation error Erase error High-speed interrupt time over error	Erase the sector specified by sector_number
Sector_Protect_Lib	ram_adr sector_number	Normal end Protect setup error Reset error Security error Already protected Address volation error Double activation error Double activation error High-speed interrupt time over error	Protect setup the sector specified by sector_number
Security_Key_Program_Lib	ram_adr key_adr	Normal end Key setup error Reset error Already secured Double activation error Double activation error High-speed interrupt time over error	Set up Security Key cord (128bit) stored in key_adr
Security_Key_Check_Lib	ram_adr key_adr	Authentication accord Double activation error Double activation error Authentication error No key setup	Authentify Security Key cord (128bit) stored in key_adr
Sector_Blank_Check_Lib	ram_adr sector_number	Blank Address volation error Double activation error Double activation error Not Blank	Confirm the sector specified by sector_number is set to Blank (All 0xFF)
Sector_Sum_Verify_Lib	ram_adr sector_number sum_data	Verify Pass Address volation error Verify Fail	Verify the sector specified by sector_number by Check Sum value of 4-byte data.
Read_Status_Lib	ram_adr (2)	None	Store Security/Protect state to the address stored by ram_adr

Command library	Argument	Return value	Function
Program_1ms_Lib	ram_adr dst_adr dst_bank bc src_adr	Normal end Writing error 1over0 error Security error Protect error Address volation error Double activation error Double activation error High-speed interrupt time over error Suspend	Write bc+1 bytes of data stored in src_adr to dst_bank, dst_adr
Erase_1ms_Lib	ram_adr sector_number	Normal end Erase error Security error Protect error Address volation error Double activation error Double activation error Erase error High-speed interrupt time over error Suspend	Erase the sector specified by sector_number

16.8 ROM Programming Service

Panasonic Corporation provides ROM programming service.
This LSI can be produced in which the arbitrary data has been written in advance.
The protect information to prevent writing/erasing errors and the security key code to prevent alteration or leakage of the program can be set.
Request it from our ROM order service.

16.8.1 ROM Data Configuration

For your ROM structure, select one among six configurations of ROM data according to your usage.
Table:16.8.1 shows ROM data configuration.

Table:16.8.1 ROM Data Configuration

		BOOT Mode Microcontroller Rewriting Method		
		Unused	Used	
Protect/ Security Function	Unused	Configuration 1	Configuration 2	Configuration 3
		MAIN	MAIN BOOT	BOOT
	Used	Configuration 4	Configuration 5	Configuration 6
		MAIN ALL"1" Protect Security	MAIN BOOT Protect Security	BOOT Protect Security

MAIN: Data for MAIN area
BOOT: Data for BOOT area
ALL"1": 0xFF padding data
Protect: Protect information
Security: Key code for security

Configuration 1

BOOT mode microcontroller rewriting method : Unused
Protect / Security function : Unused

Prepare only the data file for MAIN area.
The size of the data file for MAIN area should be adjusted to less than 128 KB [64 KB].

Configuration 2, 3

BOOT mode microcontroller rewriting method : Used
Protect / Security function : Unused

Prepare the data file for MAIN area and the data file for BOOT area. (Configuration 2)
Sector 6 [Sector 4] is used as BOOT area.
The size of the data file for MAIN area should be adjusted to less than 124 KB [60 KB].
The size of the data file for BOOT area should be adjusted to less than 4 KB.
Prepare only the data file for BOOT area. (Configuration 3)

Configuration 4

BOOT mode microcontroller rewriting method : Unused
Protect / Security function : Used

Prepare the data file for MAIN area and the data file for protect / security.
The size of the data file for MAIN area should be adjusted to less than 128 KB [64 KB].
For the data file for protect / security, refer to [Chapter 16 16.8.2 File for Protect / Security].

Configuration 5, 6

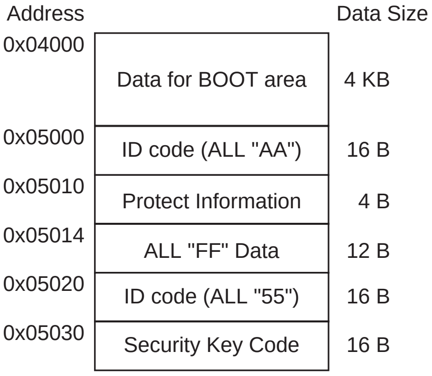
BOOT mode microcontroller rewriting method : Used
Protect / Security function : Used

Prepare the data file for MAIN area and the data file for BOOT area / protect / security. (Configuration 5)
Sector 6 [Sector 4] is used as BOOT area.
The size of the data file for MAIN area should be adjusted to less than 124 KB [60 KB].
For the data file for BOOT area / protect / security, refer to [Chapter 16 16.8.2 File for Protect / Security].
Prepare only the data file for BOOT area / protect / security. (Configuration 6)

16.8.2 File for Protect / Security

When using protect / security function, the file for protect / security should include both the data for BOOT area and for protect / security.

Figure:16.8.1 shows the configuration of the file for protect / security. Table:16.8.2 and Table:16.8.3 show the protect information.



Address		Data Size
0x04000	Data for BOOT area	4 KB
0x05000	ID code (ALL "AA")	16 B
0x05010	Protect Information	4 B
0x05014	ALL "FF" Data	12 B
0x05020	ID code (ALL "55")	16 B
0x05030	Security Key Code	16 B

Figure:16.8.1 File for Protect / Security

Table:16.8.2 Protect Information (128 KB)

Address	Protect Setup		
0x05010	bit 0	Sector 0	“0”: Enables protect “1”: Disables protect
	bit 1	Sector 1	
	bit 2	Sector 2	
	bit 3	Sector 3	
	bit 4	Sector 4	
	bit 5	Sector 5	
	bit 6	Sector 6	
	bit 7	Not used	Always set to “1”
0x05011	All bits	Not used	Always set to “0xFF”
0x05012			
0x05013			

Table:16.8.3 Protect Information (64 KB)

Address	Protect Setup		
0x05010	bit 0	Sector 0	“0”: Enables protect “1”: Disables protect
	bit 1	Sector 1	
	bit 2	Sector 2	
	bit 3	Sector 3	
	bit 4	Sector 4	
	bit 5	Not used	Always set to “1”
	bit 6		
	bit 7		
0x05011	All bits	Not used	Always set to “0xFF”
0x05012			
0x05013			

The following description shows precautions for creation of the file for protect / security.

- Compose the file for protect / security of 4 KB + 128 KB [64 KB].
- When the data for BOOT area is less than 4 KB, perform padding by "0xFF".
- When BOOT mode microcontroller rewriting method is not used, perform padding by "0xFF" to BOOT area.
- When the protect function is not used, set all the data of the protect information to "0xFF".
- When the security function is not used, set all the data of the security key code to "0xFF".
- Even when the protect or the security function is not used, allocate the ID code.

16.8.3 ROM Order Service

For the ROM order service, consult our sales offices.

Chapter 17 Appendix

17.1 Special Function Registers List

Address	Register	Bit Symbol								Page
0x03DA0	TS0TMD	-	-	-	TS0CSMD	TS0RSMD	TS0RCMD	TS0MD	TS0STEN	XV-9
		-	-	-	0	0	0	0	0	
		Tstimer 0 Control Register								
0x03DA1	TS0CKMD	-	-	-	-	TS0CK2	TS0CK1	TS0CK0	TS0PSCEN	XV-11
		-	-	-	-	0	0	0	0	
		Tstimer 0 Clock Selection Register								
0x03DA2	TS0TCHSEL	TS0IN7SEL	TS0IN6SEL	TS0IN5SEL	TS0IN4SEL	TS0IN3SEL	TS0IN2SEL	TS0IN1SEL	TS0IN0SEL	XV-12
		0	0	0	0	0	0	0	0	
		Tstimer 0 Channel Selection Register								
0x03DA3	TS0RESULT	TS0RCH7	TS0RCH6	TS0RCH5	TS0RCH4	TS0RCH3	TS0RCH2	TS0RCH1	TS0RCH0	XV-14
		0	0	0	0	0	0	0	0	
		Touch 0 Detect Interrupt Check Register								
0x03DA4	TS0ERROR	TS0ECH7	TS0ECH6	TS0ECH5	TS0ECH4	TS0ECH3	TS0ECH2	TS0ECH1	TS0ECH0	XV-16
		0	0	0	0	0	0	0	0	
		Touch 0 Detect Error Interrupt Check Register								
0x03DA5	TS0ADCNT	-	-	-	-	-	-	-	TS0ADEN	XIV-13
		-	-	-	-	-	-	-	0	
		Touch Sensor 0 A/D Control Enable Register								
0x03DB0	TS0CH0EXDATL	TS0CH0EXD7	TS0CH0EXD6	TS0CH0EXD5	TS0CH0EXD4	TS0CH0EXD3	TS0CH0EXD2	TS0CH0EXD1	TS0CH0EXD0	XV-18
		0	0	0	0	0	0	0	0	
		TS0IN0 Expected Data Register (Lower 8 bits)								
0x03DB1	TS0CH0EXDATH	-	-	-	-	-	-	TS0CH0EXD9	TS0CH0EXD8	XV-18
		-	-	-	-	-	-	0	0	
		TS0IN0 Expected Data Register (Upper 2 bits)								
0x03DB2	TS0CH1EXDATL	TS0CH1EXD7	TS0CH1EXD6	TS0CH1EXD5	TS0CH1EXD4	TS0CH1EXD3	TS0CH1EXD2	TS0CH1EXD1	TS0CH1EXD0	XV-18
		0	0	0	0	0	0	0	0	
		TS0IN1 Expected Data Register (Lower 8 bits)								
0x03DB3	TS0CH1EXDATH	-	-	-	-	-	-	TS0CH1EXD9	TS0CH1EXD8	XV-18
		-	-	-	-	-	-	0	0	
		TS0IN1 Expected Data Register (Upper 2 bits)								
0x03DB4	TS0CH2EXDATL	TS0CH2EXD7	TS0CH2EXD6	TS0CH2EXD5	TS0CH2EXD4	TS0CH2EXD3	TS0CH2EXD2	TS0CH2EXD1	TS0CH2EXD0	XV-18
		0	0	0	0	0	0	0	0	
		TS0IN2 Expected Data Register (Lower 8 bits)								
0x03DB5	TS0CH2EXDATH	-	-	-	-	-	-	TS0CH2EXD9	TS0CH2EXD8	XV-18
		-	-	-	-	-	-	0	0	
		TS0IN2 Expected Data Register (Upper 2 bits)								
0x03DB6	TS0CH3EXDATL	TS0CH3EXD7	TS0CH3EXD6	TS0CH3EXD5	TS0CH3EXD4	TS0CH3EXD3	TS0CH3EXD2	TS0CH3EXD1	TS0CH3EXD0	XV-18
		0	0	0	0	0	0	0	0	
		TS0IN3 Expected Data Register (Lower 8 bits)								

Address	Register	Bit Symbol								Page
0x03DB7	TS0CH3EX DATH	-	-	-	-	-	-	TS0CH3EXD9	TS0CH3EXD8	XV-18
		-	-	-	-	-	-	0	0	
		TS0IN3 Expected Data Register (Upper 2 bits)								
0x03DB8	TS0CH4EX DATL	TS0CH4EXD7	TS0CH4EXD6	TS0CH4EXD5	TS0CH4EXD4	TS0CH4EXD3	TS0CH4EXD2	TS0CH4EXD1	TS0CH4EXD0	XV-18
		0	0	0	0	0	0	0	0	
		TS0IN4 Expected Data Register (Lower 8 bits)								
0x03DB9	TS0CH4EX DATH	-	-	-	-	-	-	TS0CH4EXD9	TS0CH4EXD8	XV-18
		-	-	-	-	-	-	0	0	
		TS0IN4 Expected Data Register (Upper 2 bits)								
0x03DBA	TS0CH5EX DATL	TS0CH5EXD7	TS0CH5EXD6	TS0CH5EXD5	TS0CH5EXD4	TS0CH5EXD3	TS0CH5EXD2	TS0CH5EXD1	TS0CH5EXD0	XV-18
		0	0	0	0	0	0	0	0	
		TS0IN5 Expected Data Register (Lower 8 bits)								
0x03DBB	TS0CH5EX DATH	-	-	-	-	-	-	TS0CH5EXD9	TS0CH5EXD8	XV-18
		-	-	-	-	-	-	0	0	
		TS0IN5 Expected Data Register (Upper 2 bits)								
0x03DBC	TS0CH6EX DATL	TS0CH6EXD7	TS0CH6EXD6	TS0CH6EXD5	TS0CH6EXD4	TS0CH6EXD3	TS0CH6EXD2	TS0CH6EXD1	TS0CH6EXD0	XV-18
		0	0	0	0	0	0	0	0	
		TS0IN6 Expected Data Register (Lower 8 bits)								
0x03DBD	TS0CH6EX DATH	-	-	-	-	-	-	TS0CH6EXD9	TS0CH6EXD8	XV-18
		-	-	-	-	-	-	0	0	
		TS0IN6 Expected Data Register (Upper 2 bits)								
0x03DBE	TS0CH7EX DATL	TS0CH7EXD7	TS0CH7EXD6	TS0CH7EXD5	TS0CH7EXD4	TS0CH7EXD3	TS0CH7EXD2	TS0CH7EXD1	TS0CH7EXD0	XV-18
		0	0	0	0	0	0	0	0	
		TS0IN7 Expected Data Register (Lower 8 bits)								
0x03DBF	TS0CH7EX DATH	-	-	-	-	-	-	TS0CH7EXD9	TS0CH7EXD8	XV-18
		-	-	-	-	-	-	0	0	
		TS0IN7 Expected Data Register (Upper 2 bits)								
0x03DC0	TS0CH0CT DATL	TS0CH0CTD7	TS0CH0CTD6	TS0CH0CTD5	TS0CH0CTD4	TS0CH0CTD3	TS0CH0CTD2	TS0CH0CTD1	TS0CH0CTD0	XV-19
		0	0	0	0	0	0	0	0	
		TS0IN0 Measurement Data Register (Lower 8 bits)								
0x03DC1	TS0CH0CT DATH	-	-	-	-	-	-	TS0CH0CTD9	TS0CH0CTD8	XV-19
		-	-	-	-	-	-	0	0	
		TS0IN0 Measurement Data Register (Upper 2 bits)								
0x03DC2	TS0CH1CT DATL	TS0CH1CTD7	TS0CH1CTD6	TS0CH1CTD5	TS0CH1CTD4	TS0CH1CTD3	TS0CH1CTD2	TS0CH1CTD1	TS0CH1CTD0	XV-19
		0	0	0	0	0	0	0	0	
		TS0IN1 Measurement Data Register (Lower 8 bits)								
0x03DC3	TS0CH1CT DATH	-	-	-	-	-	-	TS0CH1CTD9	TS0CH1CTD8	XV-19
		-	-	-	-	-	-	0	0	
		TS0IN1 Measurement Data Register (Upper 2 bits)								
0x03DC4	TS0CH2CT DATL	TS0CH2CTD7	TS0CH2CTD6	TS0CH2CTD5	TS0CH2CTD4	TS0CH2CTD3	TS0CH2CTD2	TS0CH2CTD1	TS0CH2CTD0	XV-19
		0	0	0	0	0	0	0	0	
		TS0IN2 Measurement Data Register (Lower 8 bits)								
0x03DC5	TS0CH2CT DATH	-	-	-	-	-	-	TS0CH2CTD9	TS0CH2CTD8	XV-19
		-	-	-	-	-	-	0	0	
		TS0IN2 Measurement Data Register (Upper 2 bits)								

Address	Register	Bit Symbol								Page
0x03DC6	TS0CH3CTDATL	TS0CH3CTD7	TS0CH3CTD6	TS0CH3CTD5	TS0CH3CTD4	TS0CH3CTD3	TS0CH3CTD2	TS0CH3CTD1	TS0CH3CTD0	XV-19
		0	0	0	0	0	0	0	0	
		TS0IN3 Measurement Data Register (Lower 8 bits)								
0x03DC7	TS0CH3CTDATH	-	-	-	-	-	-	TS0CH3CTD9	TS0CH3CTD8	XV-19
		-	-	-	-	-	-	0	0	
		TS0IN3 Measurement Data Register (Upper 2 bits)								
0x03DC8	TS0CH4CTDATL	TS0CH4CTD7	TS0CH4CTD6	TS0CH4CTD5	TS0CH4CTD4	TS0CH4CTD3	TS0CH4CTD2	TS0CH4CTD1	TS0CH4CTD0	XV-19
		0	0	0	0	0	0	0	0	
		TS0IN4 Measurement Data Register (Lower 8 bits)								
0x03DC9	TS0CH4CTDATH	-	-	-	-	-	-	TS0CH4CTD9	TS0CH4CTD8	XV-19
		-	-	-	-	-	-	0	0	
		TS0IN4 Measurement Data Register (Upper 2 bits)								
0x03DCA	TS0CH5CTDATL	TS0CH5CTD7	TS0CH5CTD6	TS0CH5CTD5	TS0CH5CTD4	TS0CH5CTD3	TS0CH5CTD2	TS0CH5CTD1	TS0CH5CTD0	XV-19
		0	0	0	0	0	0	0	0	
		TS0IN5 Measurement Data Register (Lower 8 bits)								
0x03DCB	TS0CH5CTDATH	-	-	-	-	-	-	TS0CH5CTD9	TS0CH5CTD8	XV-19
		-	-	-	-	-	-	0	0	
		TS0IN5 Measurement Data Register (Upper 2 bits)								
0x03DCC	TS0CH6CTDATL	TS0CH6CTD7	TS0CH6CTD6	TS0CH6CTD5	TS0CH6CTD4	TS0CH6CTD3	TS0CH6CTD2	TS0CH6CTD1	TS0CH6CTD0	XV-19
		0	0	0	0	0	0	0	0	
		TS0IN6 Measurement Data Register (Lower 8 bits)								
0x03DCD	TS0CH6CTDATH	-	-	-	-	-	-	TS0CH6CTD9	TS0CH6CTD8	XV-19
		-	-	-	-	-	-	0	0	
		TS0IN6 Measurement Data Register (Upper 2 bits)								
0x03DCE	TS0CH7CTDATL	TS0CH7CTD7	TS0CH7CTD6	TS0CH7CTD5	TS0CH7CTD4	TS0CH7CTD3	TS0CH7CTD2	TS0CH7CTD1	TS0CH7CTD0	XV-19
		0	0	0	0	0	0	0	0	
		TS0IN7 Measurement Data Register (Lower 8 bits)								
0x03DCF	TS0CH7CTDATH	-	-	-	-	-	-	TS0CH7CTD9	TS0CH7CTD8	XV-19
		-	-	-	-	-	-	0	0	
		TS0IN7 Measurement Data Register (Upper 2 bits)								
0x03DD0	TS1TMD	-	-	-	TS1CSMD	TS1RSMD	TS1RCMD	TS1MD	TS1STEN	XV-9
		-	-	-	0	0	0	0	0	
		Tstimer 1 Control Register								
0x03DD1	TS1CKMD	-	-	-	-	TS1CK2	TS1CK1	TS1CK0	TS1PSCEN	XV-11
		-	-	-	-	0	0	0	0	
		Tstimer 1 Clock Selection Register								
0x03DD2	TS1TCHSEL	-	-	-	-	TS1IN3SEL	TS1IN2SEL	TS1IN1SEL	TS1IN0SEL	XV-12
		-	-	-	-	0	0	0	0	
		Tstimer 1 Channel Selection Register								
0x03DD3	TS1RESULT	-	-	-	-	TS1RCH3	TS1RCH2	TS1RCH1	TS1RCH0	XV-14
		-	-	-	-	0	0	0	0	
		Touch 1 Detect Interrupt Check Register								
0x03DD4	TS1ERROR	-	-	-	-	TS1ECH3	TS1ECH2	TS1ECH1	TS1ECH0	XV-16
		-	-	-	-	0	0	0	0	
		Touch 1 Detect Error Interrupt Check Register								

Address	Register	Bit Symbol								Page
0x03DE0	TS1CH0EXDATL	TS1CH0EXD7	TS1CH0EXD6	TS1CH0EXD5	TS1CH0EXD4	TS1CH0EXD3	TS1CH0EXD2	TS1CH0EXD1	TS1CH0EXD0	XV-18
		0	0	0	0	0	0	0	0	
		TS1IN0 Expected Data Register (Lower 8 bits)								
0x03DE1	TS1CH0EXDATH	-	-	-	-	-	-	TS1CH0EXD9	TS1CH0EXD8	XV-18
		-	-	-	-	-	-	0	0	
		TS1IN0 Expected Data Register (Upper 2 bits)								
0x03DE2	TS1CH1EXDATL	TS1CH1EXD7	TS1CH1EXD6	TS1CH1EXD5	TS1CH1EXD4	TS1CH1EXD3	TS1CH1EXD2	TS1CH1EXD1	TS1CH1EXD0	XV-18
		0	0	0	0	0	0	0	0	
		TS1IN1 Expected Data Register (Lower 8 bits)								
0x03DE3	TS1CH1EXDATH	-	-	-	-	-	-	TS1CH1EXD9	TS1CH1EXD8	XV-18
		-	-	-	-	-	-	0	0	
		TS1IN1 Expected Data Register (Upper 2 bits)								
0x03DE4	TS1CH2EXDATL	TS1CH2EXD7	TS1CH2EXD6	TS1CH2EXD5	TS1CH2EXD4	TS1CH2EXD3	TS1CH2EXD2	TS1CH2EXD1	TS1CH2EXD0	XV-18
		0	0	0	0	0	0	0	0	
		TS1IN2 Expected Data Register (Lower 8 bits)								
0x03DE5	TS1CH2EXDATH	-	-	-	-	-	-	TS1CH2EXD9	TS1CH2EXD8	XV-18
		-	-	-	-	-	-	0	0	
		TS1IN2 Expected Data Register (Upper 2 bits)								
0x03DE6	TS1CH3EXDATL	TS1CH3EXD7	TS1CH3EXD6	TS1CH3EXD5	TS1CH3EXD4	TS1CH3EXD3	TS1CH3EXD2	TS1CH3EXD1	TS1CH3EXD0	XV-18
		0	0	0	0	0	0	0	0	
		TS1IN3 Expected Data Register (Lower 8 bits)								
0x03DE7	TS1CH3EXDATH	-	-	-	-	-	-	TS1CH3EXD9	TS1CH3EXD8	XV-18
		-	-	-	-	-	-	0	0	
		TS1IN3 Expected Data Register (Upper 2 bits)								
0x03DF0	TS1CH0CTDATL	TS1CH0CTD7	TS1CH0CTD6	TS1CH0CTD5	TS1CH0CTD4	TS1CH0CTD3	TS1CH0CTD2	TS1CH0CTD1	TS1CH0CTD0	XV-19
		0	0	0	0	0	0	0	0	
		TS1IN0 Measurement Data Register (Lower 8 bits)								
0x03DF1	TS1CH0CTDATH	-	-	-	-	-	-	TS1CH0CTD9	TS1CH0CTD8	XV-19
		-	-	-	-	-	-	0	0	
		TS1IN0 Measurement Data Register (Upper 2 bits)								
0x03DF2	TS1CH1CTDATL	TS1CH1CTD7	TS1CH1CTD6	TS1CH1CTD5	TS1CH1CTD4	TS1CH1CTD3	TS1CH1CTD2	TS1CH1CTD1	TS1CH1CTD0	XV-19
		0	0	0	0	0	0	0	0	
		TS1IN1 Measurement Data Register (Lower 8 bits)								
0x03DF3	TS1CH1CTDATH	-	-	-	-	-	-	TS1CH1CTD9	TS1CH1CTD8	XV-19
		-	-	-	-	-	-	0	0	
		TS1IN1 Measurement Data Register (Upper 2 bits)								
0x03DF4	TS1CH2CTDATL	TS1CH2CTD7	TS1CH2CTD6	TS1CH2CTD5	TS1CH2CTD4	TS1CH2CTD3	TS1CH2CTD2	TS1CH2CTD1	TS1CH2CTD0	XV-19
		0	0	0	0	0	0	0	0	
		TS1IN2 Measurement Data Register (Lower 8 bits)								
0x03DF5	TS1CH2CTDATH	-	-	-	-	-	-	TS1CH2CTD9	TS1CH2CTD8	XV-19
		-	-	-	-	-	-	0	0	
		TS1IN2 Measurement Data Register (Upper 2 bits)								
0x03DF6	TS1CH3CTDATL	TS1CH3CTD7	TS1CH3CTD6	TS1CH3CTD5	TS1CH3CTD4	TS1CH3CTD3	TS1CH3CTD2	TS1CH3CTD1	TS1CH3CTD0	XV-19
		0	0	0	0	0	0	0	0	
		TS1IN3 Measurement Data Register (Lower 8 bits)								

Address	Register	Bit Symbol								Page
0x03DF7	TS1CH3CT DATH	-	-	-	-	-	-	TS1CH3CTD9	TS1CH3CTD8	XV-19
		-	-	-	-	-	-	0	0	
		TS1IN3 Measurement Data Register (Upper 2 bits)								
0x03E00	PWMMDL	PCRAEN	PCRBEN	INTAEN	INTBEN	DTEN	ORMD	TCEN	WAVEMD	IX-8
		0	0	0	0	0	0	0	0	
		PWM Mode Control Register Lower 8bits								
0x03E01	PWMMDH	-	INTCEN	-	Reserved	-	DTSEL	SDSELA	SDSELB	IX-9
		-	0	-	0	-	0	0	0	
		PWM Mode Control Register Upper 8bits								
0x03E02	PWMSELL	PSELN00	PSEL00	OTLVN02	OUTLV02	OUTLVN01	OUTLV01	OUTLVN00	OUTLV00	IX-11
		0	0	0	0	0	0	0	0	
		PWM Output Control Register Lower 8bits								
0x03E03	PWMSELH	-	-	-	-	PSELN02	PSEL02	PSELN01	PSEL01	IX-12
		-	-	-	-	0	0	0	0	
		PWM Output Control Register Upper 8bits								
0x03E04	PWMSETL	PMSET07	PMSET06	PMSET05	PMSET04	PMSET03	PMSET02	PMSET01	PMSET00	IX-13
		0	0	0	0	0	0	0	0	
		PWM Cycle Setting Register Lower 8bits								
0x03E05	PWMSETH	PMSET0F	PMSET0E	PMSET0D	PMSET0C	PMSETÇOB	PMSET0A	PMSET09	PMSET08	IX-13
		0	0	0	0	0	0	0	0	
		PWM Cycle Setting Register Upper 8bits								
0x03E06	TCMPAL	TCPA07	TCPA06	TCPA05	TCPA04	TCPA03	TCPA02	TCPA01	TCPA00	IX-14
		0	0	0	0	0	0	0	0	
		PWM U-Phase Comparison Setting Register Lower 8bits								
0x03E07	TCMPAH	TCPA0F	TCPA0E	TCPA0D	TCPA0C	TCPA0B	TCPA0A	TCPA09	TCPA08	IX-14
		0	0	0	0	0	0	0	0	
		PWM U-Phase Comparison Setting Register Upper 8bits								
0x03E08	TCMPBL	TCPB07	TCPB06	TCPB05	TCPB04	TCPB03	TCPB02	TCPB01	TCPB00	IX-14
		0	0	0	0	0	0	0	0	
		PWM V-Phase Comparison Setting Register Lower 8bits								
0x03E09	TCMPBH	TCPB0F	TCPB0E	TCPB0D	TCPB0C	TCPB0B	TCPB0A	TCPB09	TCPB08	IX-15
		0	0	0	0	0	0	0	0	
		PWM V-Phase Comparison Setting Register Upper 8bits								
0x03E0A	TCMPCL	TCPC07	TCPC06	TCPC05	TCPC04	TCPC03	TCPC02	TCPC01	TCPC00	IX-15
		0	0	0	0	0	0	0	0	
		PWM W-Phase Comparison Setting Register Lower 8bits								
0x03E0B	TCMPCH	TCPC0F	TCPC0E	TCPC0D	TCPC0C	TCPC0B	TCPC0A	TCPC09	TCPC08	IX-15
		0	0	0	0	0	0	0	0	
		PWM W-Phase Comparison Setting Register Upper 8bits								
0x03E0C	OUTMD	-	-	PXDTNW	PXDTW	PXDTNV	PXDTV	PXDTNU	PXDTU	IX-10
		-	-	0	0	0	0	0	0	
		PWM Output Polarity Control Register								
0x03E0D	DTMSET	DTST07	DTST06	DTST05	DTST04	DTST03	DTST02	DTST01	DTST00	IX-17
		0	0	0	0	0	0	0	0	
		PWM Dead Time Setting Register								

Address	Register	Bit Symbol								Page
0x03E0E	DTMSET1	DTST17	DTST16	DTST15	DTST14	DTST13	DTST12	DTST11	DTST10	IX-17
		0	0	0	0	0	0	0	0	
		PWM Dead Time Setting Register 1								
0x03E10	PWMBCL	PWMBC07	PWMBC06	PWMBC05	PWMBC04	PWMBC03	PWMBC02	PWMBC01	PWMBC00	IX-18
		0	0	0	0	0	0	0	0	
		PWM BC Value Read Register Lower 8bits								
0x03E11	PWMBCH	PWMBC15	PWMBC14	PWMBC13	PWMBC12	PWMBC11	PWMBC10	PWMBC9	PWMBC8	IX-18
		0	0	0	0	0	0	0	0	
		PWM BC Value Read Register Upper 8bits								
0x03E12	BCSTR	-	-	-	-	-	-	-	PWMSTR	IX-19
		-	-	-	-	-	-	-	1	
		PWM BC Status Read Register								
0x03E13	PWMOFFL	PRTANU1	PRTANU0	PRTAU1	PRTAU0	IRQSEL2	IRQSEL1	IRQSEL0	OUTEN0	IX-20
		0	0	0	0	0	0	0	0	
		PWM Pin Protection Control Register Lower 8bits								
0x03E14	PWMOFFH	PRTANW1	PRTANW0	PRTAW1	PRTAW0	PRTANV1	PRTANV0	PRTAV1	PRTAV0	IX-21
		0	0	0	0	0	0	0	0	
		PWM Pin Protection Control Register Upper 8bits								
0x03E15	IRQCULL	REGSELEDGE1	REGSELEDGE0	REGCULLCMP1	REGCULLCMP0	REGCULLUDF1	REGCULLUDF0	REGCULLOVF1	REGCULLOVF0	IX-22
		0	0	0	0	0	0	0	0	
		Interrupt Output Control Register								
0x03E16	PWMTMCNT	-	-	-	PSCSEL1	PSCSEL0	PWMCKSEL1	PWMCKSEL0	SEL_PWM_TM	IX-23
		-	-	-	0	0	0	0	0	
		PWM Timer Operation Control Register								
0x03E17	RELCTR	RELIRQA	RELIRQB	RELIRQC	RELIRQD	-	MD1CTR	RELMD1	RELMD0	IX-24
		0	0	0	0	-	0	0	0	
		4 Phases PWM Mode Register								
0x03E18	PWMODR	-	-	-	-	-	-	-	ORDER	IX-25
		-	-	-	-	-	-	-	0	
		4 Phases PWM Output Order Control Register								
0x03E19	RELSTAT	-	-	-	-	-	-	STAT1	STAT0	IX-26
		-	-	-	-	-	-	0	0	
		4 Phases PWM Output Status Register								
0x03E1A	PWMCMP1	PWMCMPL7	PWMCMPL6	PWMCMPL5	PWMCMPL4	PWMCMPL3	PWMCMPL2	PWMCMPL1	PWMCMPL0	IX-27
		0	0	0	0	0	0	0	0	
		PWM Cycle/Compare Collective Setting Register 1								
0x03E1B	PWMCMP2	PWMCMPH7	PWMCMPH6	PWMCMPH5	PWMCMPH4	PWMCMPH3	PWMCMPH2	PWMCMPH1	PWMCMPH0	IX-28
		0	0	0	0	0	0	0	0	
		PWM Cycle/Compare Collective Setting Register 2								
0x03E50	PRTKEY	PRTK7	PRTK6	PRTK5	PRTK4	PRTK3	PRTK2	PRTK1	PRTK0	XI-7
		0	0	0	0	0	0	0	0	
		Register Protect Control Register								
0x03E56	OSCLOCK	-	-	-	-	-	-	-	LOCKEN	III-13
		-	-	-	-	-	-	-	0	
		Oscillation Control Register Protect Register								

Address	Register	Bit Symbol								Page
0x03E60	TS0ATCNT0	TS0RAMDE	TS0ATACT	-	-	-	-	Reserved	TS0ATEN	XV-20
		0	0	1	1	1	1	0	0	
		Tstimer ATC 0 Control Register 0								
0x03E61	TS0ATCNT1	-	-	Reserved	Reserved	Reserved	Reserved	TS0ATIR1	TS0ATIR0	XV-21
		-	-	0	0	0	0	0	0	
		Tstimer ATC 0 Control Register 1								
0x03E62	TS0ATTRC	TS0ATTRC7	TS0ATTRC6	TS0ATTRC5	TS0ATTRC4	TS0ATTRC3	TS0ATTRC2	TS0ATTRC1	TS0ATTRC0	XV-21
		0	0	0	0	0	0	0	0	
		Tstimer ATC 0 Transfer Data Counter								
0x03E63	TS0ATRAMAPL	TS0ATRAMAP7	TS0ATRAMAP6	TS0ATRAMAP5	TS0ATRAMAP4	TS0ATRAMAP3	TS0ATRAMAP2	TS0ATRAMAP1	TS0ATRAMAP0	XV-22
		0	0	0	0	0	0	0	0	
		Tstimer 0 ATC RAM Address Pointer Lower 8 bits								
0x03E64	TS0ATRAMAPH	-	-	-	TSATRAMAP12	TSATRAMAP11	TSATRAMAP10	TSATRAMAP9	TSATRAMAP8	XV-22
		-	-	-	0	0	0	0	0	
		Tstimer ATC 0 RAM Address Pointer Upper 5 bits								
0x03E66	TS0ATREGAP	TS0ATREGAP7	TS0ATREGAP6	TS0ATREGAP5	TS0ATREGAP4	TS0ATREGAP3	TS0ATREGAP2	TS0ATREGAP1	TS0ATREGAP0	XV-23
		0	0	0	0	0	0	0	0	
		Tstimer ATC 0 Register Address Pointer								
0x03E68	TS1ATCNT0	TS1RAMDE	TS1ATACT	-	-	-	-	Reserved	TS1ATEN	XV-20
		0	0	1	1	1	1	0	0	
		Tstimer ATC 1 Control Register 0								
0x03E69	TS1ATCNT1	-	-	Reserved	Reserved	Reserved	Reserved	TS1ATIR1	TS1ATIR0	XV-21
		-	-	0	0	0	0	0	0	
		Tstimer ATC 1 Control Register 1								
0x03E6A	TS1ATTRC	TS1ATTRC7	TS1ATTRC6	TS1ATTRC5	TS1ATTRC4	TS1ATTRC3	TS1ATTRC2	TS1ATTRC1	TS1ATTRC0	XV-21
		0	0	0	0	0	0	0	0	
		Tstimer ATC 1 Transfer Data Counter								
0x03E6B	TS1ATRAMAP	TS1ATRAMAP7	TS1ATRAMAP6	TS1ATRAMAP5	TS1ATRAMAP4	TS1ATRAMAP3	TS1ATRAMAP2	TS1ATRAMAP1	TS1ATRAMAP0	XV-22
		0	0	0	0	0	0	0	0	
		Tstimer ATC 1 RAM Address Pointer Lower 8 bits								
0x03E6C	TS1ATRAMAPH	-	-	-	TSATRAMAP12	TSATRAMAP11	TSATRAMAP10	TSATRAMAP9	TSATRAMAP8	XV-22
		-	-	-	0	0	0	0	0	
		Tstimer ATC 1 RAM Address Pointer Upper 5 bits								
0x03E6E	TS1ATREGAP	TS1ATREGAP7	TS1ATREGAP6	TS1ATREGAP5	TS1ATREGAP4	TS1ATREGAP3	TS1ATREGAP2	TS1ATREGAP1	TS1ATREGAP0	XV-23
		0	0	0	0	0	0	0	0	
		Tstimer ATC 1 Register Address Pointer								
0x03E70	P0OUT	P0OUT7	P0OUT6	P0OUT5	P0OUT4	P0OUT3	P0OUT2	P0OUT1	P0OUT0	V-9
		x	x	x	x	x	x	x	x	
		Port 0 Output Register								
0x03E72	P2OUT	P2OUT7	P2OUT6	P2OUT5	P2OUT4	P2OUT3	P2OUT2	P2OUT1	P2OUT0	V-19
		1	x	x	x	x	x	x	x	
		Port 2 Output Register								

Address	Register	Bit Symbol								Page
0x03E73	P3OUT	-	-	P3OUT5	P3OUT4	P3OUT3	-	-	-	V-26
		-	-	x	x	x	-	-	-	
		Port 3 Output Register								
0x03E74	P4OUT	P4OUT7	P4OUT6	P4OUT5	P4OUT4	P4OUT3	-	-	-	V-38
		x	x	x	x	x	-	-	-	
		Port 4 Output Register								
0x03E75	P5OUT	P5OUT7	P5OUT6	P5OUT5	P5OUT4	P5OUT3	P5OUT2	P5OUT1	P5OUT0	V-46
		x	x	x	x	x	x	x	x	
		Port 5 Output Register								
0x03E76	P6OUT	P6OUT7	P6OUT6	P6OUT5	P6OUT4	P6OUT3	P6OUT2	-	-	V-70
		x	x	x	x	x	x	-	-	
		Port 6 Output Register								
0x03E77	P7OUT	P7OUT7	P7OUT6	P7OUT5	P7OUT4	P7OUT3	P7OUT2	P7OUT1	P7OUT0	V-79
		x	x	x	x	x	x	x	x	
		Port 7 Output Register								
0x03E78	P8OUT	P8OUT7	P8OUT6	P8OUT5	P8OUT4	P8OUT3	P8OUT2	P8OUT1	P8OUT0	V-86
		x	x	x	x	x	x	x	x	
		Port 8 Output Register								
0x03E79	P9OUT	-	-	-	P9OUT4	P9OUT3	P9OUT2	P9OUT1	P9OUT0	V-92
		-	-	-	x	x	x	x	x	
		Port 9 Output Register								
0x03E7A	PAOUT	PAOUT7	PAOUT6	PAOUT5	PAOUT4	PAOUT3	PAOUT2	PAOUT1	PAOUT0	V-108
		x	x	x	x	x	x	x	x	
		Port A Output Register								
0x03E7B	PBOUT	-	-	-	-	PBOUT3	PBOUT2	PBOUT1	PBOUT0	V-117
		-	-	-	-	x	x	x	x	
		Port B Output Register								
0x03E80	P0IN	P0IN7	P0IN6	P0IN5	P0IN4	P0IN3	P0IN2	P0IN1	P0IN0	V-9
		x	x	x	x	x	x	x	x	
		Port 0 Input Register								
0x03E82	P2IN	P2IN7	P2IN6	P2IN5	P2IN4	P2IN3	P2IN2	P2IN1	P2IN0	V-19
		1	x	x	x	x	x	x	x	
		Port 2 Input Register								
0x03E83	P3IN	-	-	P3IN5	P3IN4	P3IN3	-	-	-	V-26
		-	-	x	x	x	-	-	-	
		Port 3 Input Register								
0x03E84	P4IN	P4IN7	P4IN6	P4IN5	P4IN4	P4IN3	-	-	-	V-38
		x	x	x	x	x	-	-	-	
		Port 4 Input Register								
0x03E85	P5IN	P5IN7	P5IN6	P5IN5	P5IN4	P5IN3	P5IN2	P5IN1	P5IN0	V-46
		x	x	x	x	x	x	x	x	
		Port 5 Input Register								
0x03E86	P6IN	P6IN7	P6IN6	P6IN5	P6IN4	P6IN3	P6IN2	-	-	V-70
		x	x	x	x	x	x	-	-	
		Port 6 Input Register								

Address	Register	Bit Symbol								Page
0x03E87	P7IN	P7IN7	P7IN6	P7IN5	P7IN4	P7IN3	P7IN2	P7IN1	P7IN0	V-79
		x	x	x	x	x	x	x	x	
		Port 7 Input Register								
0x03E88	P8IN	P8IN7	P8IN6	P8IN5	P8IN4	P8IN3	P8IN2	P8IN1	P8IN0	V-86
		x	x	x	x	x	x	x	x	
		Port 8 Input Register								
0x03E89	P9IN	-	-	-	P9IN4	P9IN3	P9IN2	P9IN1	P9IN0	V-93
		-	-	-	x	x	x	x	x	
		Port 9 Input Register								
0x03E8A	PAIN	PAIN7	PAIN6	PAIN5	PAIN4	PAIN3	PAIN2	PAIN1	PAIN0	V-108
		x	x	x	x	x	x	x	x	
		Port A Input Register								
0x03E8B	PBIN	-	-	-	-	PBIN3	PBIN2	PBIN1	PBIN0	V-117
		-	-	-	-	x	x	x	x	
		Port B Input Register								
0x03E90	P0DIR	P0DIR7	P0DIR6	P0DIR5	P0DIR4	P0DIR3	P0DIR2	P0DIR1	P0DIR0	V-10
		0	0	0	0	0	0	0	0	
		Port 0 Direction Control Register								
0x03E92	P2DIR	-	P2DIR6	P2DIR5	P2DIR4	P2DIR3	P2DIR2	P2DIR1	P2DIR0	V-19
		-	0	0	0	0	0	0	0	
		Port 2 Direction Control Register								
0x03E93	P3DIR	-	P3DIR6	P3DIR5	P3DIR4	P3DIR3	P3DIR2	P3DIR1	P3DIR0	V-27
		-	0	0	0	0	0	0	0	
		Port 3 Direction Control Register								
0x03E94	P4DIR	P4DIR7	P4DIR6	P4DIR5	P4DIR4	P4DIR3	-	-	-	V-39
		0	0	0	0	0	-	-	-	
		Port 4 Direction Control Register								
0x03E95	P5DIR	P5DIR7	P5DIR6	P5DIR5	P5DIR4	P5DIR3	P5DIR2	P5DIR1	P5DIR0	V-46
		0	0	0	0	0	0	0	0	
		Port 5 Direction Control Register								
0x03E96	P6DIR	P6DIR7	P6DIR6	P6DIR5	P6DIR4	P6DIR3	P6DIR2	-	-	V-70
		0	0	0	0	0	0	-	-	
		Port 6 Direction Control Register								
0x03E97	P7DIR	P7DIR7	P7DIR6	P7DIR5	P7DIR4	P7DIR3	P7DIR2	P7DIR1	P7DIR0	V-79
		0	0	0	0	0	0	0	0	
		Port 7 Direction Control Register								
0x03E98	P8DIR	P8DIR7	P8DIR6	P8DIR5	P8DIR4	P8DIR3	P8DIR2	P8DIR1	P8DIR0	V-86
		0	0	0	0	0	0	0	0	
		Port 8 Direction Control Register								
0x03E99	P9DIR	-	-	-	P9DIR4	P9DIR3	P9DIR2	P9DIR1	P9DIR0	V-94
		-	-	-	0	0	0	0	0	
		Port 9 Direction Control Register								
0x03E9A	PADIR	PADIR7	PADIR6	PADIR5	PADIR4	PADIR3	PADIR2	PADIR1	PADIR0	V-108
		0	0	0	0	0	0	0	0	
		Port A Direction Control Register								

Address	Register	Bit Symbol								Page
0x03E9B	PBDIR	-	-	-	-	PBDIR3	PBDIR2	PBDIR1	PBDIR0	V-117
		-	-	-	-	0	0	0	0	
		Port B Direction Control Register								
0x03EA0	P0PLU	P0PLU7	P0PLU6	P0PLU5	P0PLU4	P0PLU3	P0PLU2	P0PLU1	P0PLU0	V-11
		0	0	0	0	0	0	0	0	
		Port 0 Pull-up Resistor Control Register								
0x03EA2	P2PLU	-	P2PLU6	P2PLU5	P2PLU4	P2PLU3	P2PLU2	P2PLU1	P2PLU0	V-20
		-	0	0	0	0	0	0	0	
		Port 2 Pull-up Resistor Control Resister								
0x03EA3	P3PLUD	-	-	P3PLUD5	P3PLUD4	P3PLUD3	-	-	-	V-27
		-	-	0	0	0	-	-	-	
		Port 3 Pull-up/pull-down Resitor Control Register								
0x03EA4	P4PLU	P4PLU7	P4PLU6	P4PLU5	P4PLU4	P4PLU3	-	-	-	V-39
		0	0	0	0	0	-	-	-	
		Port 4 Pull-up Resistor Control Register								
0x03EA5	P5PLU(D)	P5PLU(D)7	P5PLU(D)6	P5PLU(D)5	P5PLU(D)4	P5PLU(D)3	P5PLU(D)2	P5PLU(D)1	P5PLU(D)0	V-47
		0	0	0	0	0	0	0	0	
		Port 5 Pull-up(/pull-down) Resistor Control Register								
0x03EA6	P6PLU	P6PLU7	P6PLU6	P6PLU5	P6PLU4	P6PLU3	P6PLU2	-	-	V-71
		0	0	0	0	0	0	-	-	
		Port 6 Pull-up Resistor Control Register								
0x03EA7	P7PLU	P7PLU7	P7PLU6	P7PLU5	P7PLU4	P7PLU3	P7PLU2	P7PLU1	P7PLU0	V-80
		0	0	0	0	0	0	0	0	
		Port 7 Pull-up Resistor Control Register								
0x03EA8	P8PLU	P8PLU7	P8PLU6	P8PLU5	P8PLU4	P8PLU3	P8PLU2	P8PLU1	P8PLU0	V-87
		0	0	0	0	0	0	0	0	
		Port 8 Pull-up Resistor Control Register								
0x03EA9	P9PLUD	-	-	-	P9PLUD4	P9PLUD3	P9PLUD2	P9PLUD1	P9PLUD0	V-95
		-	-	-	0	0	0	0	0	
		Port 9 Pull-up/pull-down Resistor Control Register								
0x03EAA	PAPLU	PAPLU7	PAPLU6	PAPLU5	PAPLU4	PAPLU3	PAPLU2	PAPLU1	PAPLU0	V-109
		0	0	0	0	0	0	0	0	
		Port A Pull-up Resistor Control Register								
0x03EAB	PBPLUD	-	-	-	-	PBPLUD3	PBPLUD2	PBPLUD1	PBPLUD0	V-118
		-	-	-	-	0	0	0	0	
		Port B Pull-up/pull-down Resistor Control Register								
0x03EAF	SELUD	-	-	SELUD5	-	SELUD3	-	-	-	V-29
		-	-	0	-	0	-	-	-	
		Pull-up/pull-down Resistor selection Register								
0x03EB0	P0OMD1	-	-	-	P0OMD14	P0OMD13	P0OMD12	P0OMD11	-	V-12
		-	-	-	0	0	0	0	-	
		Port 0 Output Mode Regisiter 1								
0x03EB5	P5OMD	P5OMD7	P5OMD6	-	-	-	-	-	-	V-47
		0	0	-	-	-	-	-	-	
		Port 5 Output Mode Regisiter								

Address	Register	Bit Symbol								Page
0x03EB6	P6OMD	-	-	-	-	P6OMD3	P6OMD2	-	-	V-71
		-	-	-	-	0	0	-	-	
		Port 6 Output Mode Regisiter								
0x03EB8	P8OMD	P8OMD7	P8OMD6	P8OMD5	P8OMD4	P8OMD3	P8OMD2	P8OMD1	P8OMD0	V-87
		0	0	0	0	0	0	0	0	
		Port 8 Output Mode Regisiter								
0x03EBA	PAOMD	PAOMD7	PAOMD6	PAOMD5	Reserved	PAOMD3	PAOMD2	PAOMD1	PAOMD0	V-109
		0	0	0	0	0	0	0	0	
		Port A Output Mode Regisiter								
0x03EBF	SELUD2	-	-	-	-	SELUDB	-	SELUD9	-	V-29
		-	-	-	-	0	-	0	-	
		Pull-up/pull-down Resistor Selection Register 2								
0x03EC0	P0OMD2	-	-	-	P0OMD24	-	-	-	-	V-12
		-	-	-	0	-	-	-	-	
		Port 0 Output Mode Regisiter 2								
0x03EC3	P3IMD	-	-	-	-	P3IMD3	-	-	-	V-28
		-	-	-	-	0	-	-	-	
		Port 3 Input Mode Register								
0x03EC5	P5IMD	P5IMD7	P5IMD6	P5IMD5	-	-	-	-	-	V-48
		0	0	0	-	-	-	-	-	
		Port 5 Input Mode Register								
0x03EC9	P9IMD	-	-	-	P9IMD4	P9IMD3	P9IMD2	-	-	V-96
		-	-	-	0	0	0	-	-	
		Port 9 Input Mode Register								
0x03ECA	PAIMD	PAIMD7	PAIMD6	PAIMD5	PAIMD4	PAIMD3	PAIMD2	PAIMD1	PAIMD0	V-110
		0	0	0	0	0	0	0	0	
		Port A Input Mode Register								
0x03ECB	PBIMD	-	-	-	-	PBIMD3	PBIMD2	PBIMD1	PBIMD0	V-118
		-	-	-	-	0	0	0	0	
		Port B Input Mode Register								
0x03EE0	LEDCNT	LEDCNT7	LEDCNT6	LEDCNT5	LEDCNT4	LEDCNT3	LEDCNT2	LEDCNT1	LEDCNT0	V-111
		0	0	0	0	0	0	0	0	
		Port LED Control Register								
0x03EF0	P0ODC	-	-	-	P0ODC4	P0ODC3	-	-	-	V-13
		-	-	-	0	0	-	-	-	
		Port 0 Nch Open-drain Control Register								
0x03EF3	P3ODC	-	-	-	P3ODC4	P3ODC3	-	-	-	V-28
		-	-	-	0	0	-	-	-	
		Port 3 Nch Open-drain Control Register								
0x03EF4	P4ODC	-	-	P4ODC5	-	P4ODC3	-	-	-	V-40
		-	-	0	-	0	-	-	-	
		Port 4 Nch Open-drain Control Register								
0x03EF5	P5ODC	-	-	-	-	-	P5ODC2	-	P5ODC0	V-48
		-	-	-	-	-	0	-	0	
		Port 5 Nch Open-drain Control Register								

Address	Register	Bit Symbol								Page
0x03EF6	P6ODC	P6ODC7	-	P6ODC5	-	-	-	-	-	V-72
		0	-	0	-	-	-	-	-	
		Port 6 Nch Open-drain Control Register								
0x03EF7	P7ODC	P7ODC7	-	P7ODC5	-	-	P7ODC2	P7ODC1	-	V-80
		0	-	0	-	-	0	0	-	
		Port 7 Nch Open-drain Control Register								
0x03F00	CPUM	Reserved	OSCSEL1	OSCSEL0	OSCDBL	STOP	HALT	OSC1	OSC0	III-16
		0	0	1	0	0	0	0	0	
		CPU Mode Control Register								
0x03F01	MEMCTR	IOW1	IOW0	IVBM	Reserved	Reserved	IRWE	Reserved	Reserved	II-35
		1	1	0	0	1	0	1	1	
		Memory Control Register								
0x03F02	WDCTR	-	-	Reserved	Reserved	Reserved	WDTS1	WDTS0	WDEN	XI-5
		-	-	0	0	0	1	1	0	
		Watchdog Timer Control Register								
0x03F03	DLYCTR	BUZOE	BUZS2	BUZS1	BUZS0	DLYS1	DLYS0	-	-	XI-6
		0	0	0	0	0	1	-	-	
		Oscillation Stabilization Wait Control Register								
0x03F06	HANDSHAKE	-	-	-	-	-	ROMHND	-	-	III-18
		-	-	-	-	-	1	-	-	
		Internal ROM Access Method Control Register								
0x03F07	AUCTR	-	-	-	-	-	AUDIVU	AUMUL	AUMULU	II-38
		-	-	-	-	-	0	0	0	
		Expanded Calulation Control Register								
0x03F0A	SBNKR	-	-	-	-	SBA3	SBA2	SBA1	SBA0	II-22
		-	-	-	-	0	0	0	0	
		Bank Register for Source Address								
0x03F0B	DBNKR	-	-	-	-	DBA3	DBA2	DBA1	DBA0	II-23
		-	-	-	-	0	0	0	0	
		Bank Register for Destination Address								
0x03F10	SC0SEL	SBO0SEL	SC0BRP2	SC0BRP1	SC0BRP0	OSL0	SC0SEL2	SC0SEL1	SC0SEL0	XIII-15
		0	0	0	0	0	0	0	0	
		Serial 0 I/O Pin Switching Control Register								
0x03F11	SC0MD0	SC0CE1	SC0SSC	SC0CTM	SC0DIR	SC0STE	SC0LNG2	SC0LNG1	SC0LNG0	XIII-23
		0	0	0	0	0	1	1	1	
		Serial Interface 0 Mode Register 0								
0x03F12	SC0MD1	SC0IOM	SC0SBTS	SC0SBIS	SC0SBOS	SC0CKM	SC0MST	SC0DIV	SC0CMD	XIII-23
		0	0	0	0	0	0	0	0	
		Serial Interface 0 Mode Register 1								
0x03F13	SC0MD2	SC0FM1	SC0FM0	SC0PM1	SC0PM0	SC0NPE	-	SC0BRKF	SC0BRKE	XIII-24
		0	0	0	0	0	-	0	0	
		Serial Interface 0 Mode Register 2								
0x03F14	SC0MD3	SC0FDC1	SC0FDC0	-	-	SC0PSCE	SC0PSC2	SC0PSC1	SC0PSC0	XIII-25
		0	0	-	-	0	0	0	0	
		Serial Interface 0 Mode Register 3								

Address	Register	Bit Symbol								Page
0x03F15	SC0STR	SC0TBSY	SC0RBSY	SC0TEMP	SC0REMP	SC0FEF	SC0PEK	SC0ORE	SC0ERE	XIII-26
		0	0	0	0	0	0	0	0	
		Serial Interface 0 Status Register								
0x03F16	RXBUF0	RXBUF07	RXBUF06	RXBUF05	RXBUF04	RXBUF03	RXBUF02	RXBUF01	RXBUF00	XIII-21
		x	x	x	x	x	x	x	x	
		Serial Interface 0 Reception Data Buffer								
0x03F17	TXBUF0	TXBUF07	TXBUF06	TXBUF05	TXBUF04	TXBUF03	TXBUF02	TXBUF01	TXBUF00	XIII-21
		x	x	x	x	x	x	x	x	
		Serial Interface 0 Transmission Data Buffer								
0x03F19	RSTFACT	-	-	-	-	-	-	-	RSTMON	II-51
		-	-	-	-	-	-	-	0	
		Reset Factor Determination Register								
0x03F1A	OSCCNT	HOSCDIVSEL2	HOSCDIVSEL1	HOSCDIVSEL0	HOSCDIVSTOP	-	-	HOSCSEL	HOSCCNT	III-11
		0	0	0	0	-	-	0	0	
		External High-speed Oscillation Control Register								
0x03F1B	RCCNT	-	-	-	-	-	-	Reserved	RCON	III-11
		-	-	-	-	-	-	1	0	
		Internal High-speed Oscillation Control Register								
0x03F1C	OSCSCNT	-	-	-	-	-	-	-	SOSCCNT	III-12
		-	-	-	-	-	-	-	0	
		External Low-speed Oscillation Control Register								
0x03F1E	EDGDT	EDGSEL7	-	-	-	EDGSEL3	EDGSEL2	-	EDGSEL0	IV-46
		0	-	-	-	0	0	-	0	
		Both Edges Intrrupt Control Register								
0x03F1F	PLLCNT	PLLCK3	PLLCK2	PLLCK1	PLLCK0	-	-	PLLEN	PLLSTART	III-14
		0	0	0	0	-	-	0	0	
		Clock Multipication Circuit Control Register								
0x03F20	SC1SEL	SBO1SEL	SC1BRP2	SC1BRP1	SC1BRP0	OSL1	SC1SEL2	SC1SEL1	SC1SEL0	XIII-17
		0	0	0	0	0	0	0	0	
		Serial 1 I/O Switching Control Register								
0x03F21	SC1MD0	SC1CE1	SC1SSC	SC1CTM	SC1DIR	SC1STE	SC1LNG2	SC1LNG1	SC1LNG0	XIII-22
		0	0	0	0	0	1	1	1	
		Serial interface 1 Mode Register 0								
0x03F22	SC1MD1	SC1IOM	SC1SBTS	SC1SBIS	SC1SBOS	SC1CKM	SC1MST	SC1DIV	SC1CMD	XIII-23
		0	0	0	0	0	0	0	0	
		Serial Interface 1 Mode Register 1								
0x03F23	SC1MD2	SC1FM1	SC1FM0	SC1PM1	SC1PM0	SC1NPE	-	SC1BRKF	SC1BRKE	XIII-24
		0	0	0	0	0	-	0	0	
		Serial Interface 1 Mode Register 2								
0x03F24	SC1MD3	SC1FDC1	SC1FDC0	-	-	SC1PSCE	SC1PSC2	SC1PSC1	SC1PSC1	XIII-25
		0	0	-	-	0	0	0	0	
		Serial Interface 1 Mode Register 3								
0x03F25	SC1STR	SC1TBSY	SC1RBSY	SC1TEMP	SC1REMP	SC1FEF	SC1PEK	SC1ORE	SC1ERE	XIII-26
		0	0	0	0	0	0	0	0	
		Serial Interface 1 Status Register								

Address	Register	Bit Symbol								Page
0x03F26	RXBUF1	RXBUF17	RXBUF16	RXBUF15	RXBUF14	RXBUF13	RXBUF12	RXBUF11	RXBUF10	XIII-21
		x	x	x	x	x	x	x	x	
		Serial Interface 1 Reception Data Buffer								
0x03F27	TXBUF1	TXBUF17	TXBUF16	TXBUF15	TXBUF14	TXBUF13	TXBUF12	TXBUF11	TXBUF10	XIII-21
		x	x	x	x	x	x	x	x	
		Serial Interface 1 Transmission Data Buffer								
0x03F28	SCINTSEL	-	-	-	-	SC4IGC	SC2IGC	SC1IGC	SC0IGC	XIII-27
		-	-	-	-	0	0	0	0	
		Clock Synchrounous Serial Reception Interrupt Control Register								
0x03F30	SC2SEL	SBO2SEL	SC2BRP2	SC2BRP1	SC2BRP0	-	SC2SEL2	SC2SEL1	SC2SEL0	XIII-18
		0	0	0	0	-	0	0	0	
		Serial 2 I/O Switching Control Register								
0x03F31	SC2MD0	SC2CE1	SC2SSC	SC2CTM	SC2DIR	SC2STE	SC2LNG2	SC2LNG1	SC2LNG0	XIII-22
		0	0	0	0	0	1	1	1	
		Serial interface 2 Mode Register 0								
0x03F32	SC2MD1	SC2IOM	SC2SBTS	SC2SBIS	SC2SBOS	SC2CKM	SC2MST	SC2DIV	SC2CMD	XIII-23
		0	0	0	0	0	0	0	0	
		Seria Interface 2 Mode Register 1								
0x03F33	SC2MD2	SC2FM1	SC2FM0	SC2PM1	SC2PM0	SC2NPE	-	SC2BRKF	SC2BRKE	XIII-24
		0	0	0	0	0	-	0	0	
		Serial Interface 2 Mode Register 2								
0x03F34	SC2MD3	SC2FDC1	SC2FDC0	-	-	SC2PSCE	SC2PSC2	SC2PSC1	SC2PSC1	XIII-25
		0	0	-	-	0	0	0	0	
		Serial Interface 2 Mode Register 3								
0x03F35	SC2STR	SC2TBSY	SC2RBSY	SC2TEMP	SC2REMP	SC2FEF	SC2PEK	SC2ORE	SC2ERE	XIII-26
		0	0	0	0	0	0	0	0	
		Serial Interface 2 Status Register								
0x03F36	RXBUF2	RXBUF27	RXBUF26	RXBUF25	RXBUF24	RXBUF23	RXBUF22	RXBUF21	RXBUF20	XIII-21
		x	x	x	x	x	x	x	x	
		Serial Interface 2 Reception Data Buffer								
0x03F37	TXBUF2	TXBUF27	TXBUF26	TXBUF25	TXBUF24	TXBUF23	TXBUF22	TXBUF21	TXBUF20	XIII-21
		x	x	x	x	x	x	x	x	
		Serial Interface 2 Transmission Data Buffer								
0x03F3E	KEYT3_1IMD	KEYT3SEL	-	-	-	KEYT3_1EN3	KEYT3_1EN2	KEYT3_1EN1	KEYT3_1EN0	IV-48
		0	-	-	-	0	0	0	0	
		Key Interrupt Control Register 1								
0x03F3F	KEYT3_2IMD	-	-	-	-	KEYT3_2EN3	KEYT3_2EN2	KEYT3_2EN1	KEYT3_2EN0	IV-49
		-	-	-	-	0	0	0	0	
		Key Interrupt Control Register 2								
0x03F4E	IRQEXPEN	Reserved	IRQEXPEN6	IRQEXPEN5	IRQEXPEN4	IRQEXPEN3	IRQEXPEN2	IRQEXPEN1	IRQEXPEN0	IV-36
		0	0	0	0	0	0	0	0	
		Peripheral Function Group Interrupt Input Enable Register								
0x03F4F	IRQEXPDT	Reserved	IRQEXPDT6	IRQEXPDT5	IRQEXPDT4	IRQEXPDT3	IRQEXPDT2	IRQEXPDT1	IRQEXPDT0	IV-37
		0	0	0	0	0	0	0	0	
		Peripheral Function Group Interrupt Input Factor Retention Register								

Address	Register	Bit Symbol								Page
0x03F50	SC4MD0	SC4CE1	SC4SSC	SC4DEM	SC4DIR	SC4STE	SC4LNG2	SC4LNG1	SC4LNG0	XIII-29
		0	0	0	0	0	1	1	1	
		Serial Interface 4 Mode Register 0								
0x03F51	SC4MD1	SC4IOM	SC4SBTS	SC4SBIS	SC4SBOS	-	SC4MST	-	SC4CTM	XIII-30
		0	0	0	0	-	0	-	0	
		Serial Interface 4 Mode Register 1								
0x03F52	SC4MD2	SC4FDC1	SC4FDC0	-	-	SC4PSCE	SC4PSC4	SC4PSC4	SC4PSC4	XIII-31
		0	0	-	-	0	0	0	0	
		Serial Interface 4 Mode Register 2								
0x03F53	SC4MD3	Reserved	SC4ADM	SC4STPC	SC4TMD	SC4REX	SC4CMD	SC4ACKS	SC4ACKO	XIII-32
		0	0	0	0	0	0	0	0	
		Serial Interface 4 Mode Register 3								
0x03F54	SC4AD0	SC4ADR7	SC4ADR6	SC4ADR5	SC4ADR4	SC4ADR3	SC4ADR2	SC4ADR1	SC4ADR0	XIII-33
		0	0	0	0	0	0	0	0	
		Serial Interface 4 Address Set Register 0								
0x03F55	SC4AD1	-	-	-	-	-	-	SC4ADR9	SC4ADR8	XIII-33
		-	-	-	-	-	-	0	0	
		Serial Interface 4 Address Set Register 1								
0x03F56	SC4STR0	SC4BSY	-	SC4TEMP	SC4REMP	-	-	-	SC4ORE	XIII-34
		0	-	0	0	-	-	-	0	
		Serial Interface 4 Status Register 0								
0x03F57	SC4STR1	SC4WRS	SC4ABT_LST	SC4ADD_ACC	SC4STRT	SC4BUSBSY	SC4IICBSY	SC4GCALL	SC4DATA_ERR	XIII-35
		0	0	0	0	0	0	0	0	
		Serial Interface 4 Status Register 1								
0x03F58	RXBUF4	RXBUF47	RXBUF46	RXBUF45	RXBUF44	RXBUF43	RXBUF42	RXBUF41	RXBUF40	XIII-28
		x	x	x	x	x	x	x	x	
		Serial Interface 4 Reception Data Buffer								
0x03F59	TXBUF4	TXBUF47	TXBUF46	TXBUF45	TXBUF44	TXBUF43	TXBUF42	TXBUF41	TXBUF40	XIII-28
		x	x	x	x	x	x	x	x	
		Serial Interface 4 Transmission Data Buffer								
0x03F5A	SC4SEL	Reserved	SC4BRP2	SC4BRP1	SC4BRP0	OSL4	SC4SEL2	SC4SEL1	SC4SEL0	XIII-19
		0	0	0	0	0	0	0	0	
		Serial 4 I/O Pin Switching Control Register								
0x03F5E	SBT_EDG	-	-	-	STB_EDG_IRQ4	STB_EDG_IRQ3	STB_EDG_IRQ2	STB_EDG_IRQ1	STB_EDG_IRQ0	IV-61
		-	-	-	0	0	0	0	0	
		Edge Interrupt in STANDBY Control Register								
0x03F5F	STB_MSK	-	-	-	STB_MSK_IRQ4	STB_MSK_IRQ3	STB_MSK_IRQ2	STB_MSK_IRQ1	STB_MSK_IRQ0	IV-51
		-	-	-	0	0	0	0	0	
		Interrupt in STANDBY Control Register								
0x03F60	TM0BC	TM0BC7	TM0BC6	TM0BC5	TM0BC4	TM0BC3	TM0BC2	TM0BC1	TM0BC0	VI-11
		0	0	0	0	0	0	0	0	
		Timer 0 Binary Counter								
0x03F61	TM1BC	TM1BC7	TM1BC6	TM1BC5	TM1BC4	TM1BC3	TM1BC2	TM1BC1	TM1BC0	VI-11
		0	0	0	0	0	0	0	0	
		Timer 1 Binary Counter								

Address	Register	Bit Symbol								Page
0x03F62	TM0OC	TM0OC7	TM0OC6	TM0OC5	TM0OC4	TM0OC3	TM0OC2	TM0OC1	TM0OC0	VI-11
		x	x	x	x	x	x	x	x	
		Timer 0 Compare Register								
0x03F63	TM1OC	TM1OC7	TM1OC6	TM1OC5	TM1OC4	TM1OC3	TM1OC2	TM1OC1	TM1OC0	VI-11
		x	x	x	x	x	x	x	x	
		Timer 1 Compare Register								
0x03F64	TM0MD	-	TM0POP	TM0MOD	TM0PWM	TM0EN	TM0CK2	TM0CK1	TM0CK0	VI-12
		-	0	0	0	0	0	0	0	
		Timer 0 Mode Register								
0x03F65	TM1MD	Reserved	Reserved	TM1CAS1	TM1CAS0	TM1EN	TM1CK2	TM1CK1	TM1CK0	VI-13
		0	0	0	0	0	0	0	0	
		Timer 1 Mode Register								
0x03F66	CK0MD	-	TM0ADD1	TM0ADD0	TM0ADDEN	TM0PSC2	TM0PSC1	TM0PSC0	TM0BAS	VI-9
		-	0	0	0	0	0	0	0	
		Timer 0 Prescaler Selection Register								
0x03F67	CK1MD	-	-	-	-	TM1PSC2	TM1PSC1	TM1PSC0	TM1BAS	VI-10
		-	-	-	-	0	0	0	0	
		Timer 1 Prescaler Selection Register								
0x03F68	TM2BC	TM2BC7	TM2BC6	TM2BC5	TM2BC4	TM2BC3	TM2BC2	TM2BC1	TM2BC0	VI-11
		0	0	0	0	0	0	0	0	
		Timer 2 Binary Counter								
0x03F69	TM3BC	TM3BC7	TM3BC6	TM3BC5	TM3BC4	TM3BC3	TM3BC2	TM3BC1	TM3BC0	VI-11
		0	0	0	0	0	0	0	0	
		Timer 3 Binary Counter								
0x03F6A	TM2OC	TM2OC7	TM2OC6	TM2OC5	TM2OC4	TM2OC3	TM2OC2	TM2OC1	TM2OC0	VI-11
		x	x	x	x	x	x	x	x	
		Timer 2 Compare Register								
0x03F6B	TM3OC	TM3OC7	TM3OC6	TM3OC5	TM3OC4	TM3OC3	TM3OC2	TM3OC1	TM3OC0	VI-11
		x	x	x	x	x	x	x	x	
		Timer 3 Compare Register								
0x03F6C	TM2MD	-	TM2POP	TM2MOD	TM2PWM	TM2EN	TM2CK2	TM2CK1	TM2CK0	VI-14
		-	0	0	0	0	0	0	0	
		Timer 2 Mode Register								
0x03F6D	TM3MD	-	-	-	TM3CAS	TM3EN	TM3CK2	TM3CK1	TM3CK0	VI-16
		-	-	-	0	0	0	0	0	
		Timer 3 Mode Register								
0x03F6E	CK2MD	-	TM2ADD1	TM2ADD0	TM2ADDEN	TM2PSC2	TM2PSC1	TM2PSC0	TM2BAS	VI-9
		-	0	0	0	0	0	0	0	
		Timer 2 Prescaler Selection Register								
0x03F6F	CK3MD	-	-	-	-	TM3PSC2	TM3PSC1	TM3PSC0	TM3BAS	VI-10
		-	-	-	-	0	0	0	0	
		Timer 3 Prescaler Selection Register								
0x03F71	TMABC	TMABC7	TMABC6	TMABC5	TMABC4	TMABC3	TMABC2	TMABC1	TMABC0	VII-5
		0	0	0	0	0	0	0	0	
		Timer A Binary Counter								

Address	Register	Bit Symbol								Page
0x03F73	TMAOC	TMAOC7	TMAOC6	TMAOC5	TMAOC4	TMAOC3	TMAOC2	TMAOC1	TMAOC0	VII-5
		x	x	x	x	x	x	x	x	
		Timer A Compare Register								
0x03F75	TMAMD1	-	-	-	-	TMAEN	TMACK2	TMACK1	TMACK0	VII-6
		-	-	-	-	0	0	0	0	
		Timer A Mode Register 1								
0x03F77	TMAMD2	Resereved	PSCEN	-	-	-	-	-	-	VII-6
		0	0	-	-	-	-	-	-	
		Timer A Mode Register 2								
0x03F78	TM6BC	TM6BC7	TM6BC6	TM6BC5	TM6BC4	TM6BC3	TM6BC2	TM6BC1	TM6BC0	X-7
		0	0	0	0	0	0	0	0	
		Timer 6 Binary Counter								
0x03F79	TM6OC	TM6OC7	TM6OC6	TM6OC5	TM6OC4	TM6OC3	TM6OC2	TM6OC1	TM6OC0	X-7
		x	x	x	x	x	x	x	x	
		Timer 6 Compare Register								
0x03F7A	TM6MD	TM6CLRS	TM6IR2	TM6IR1	TM6IR0	TM6CK3	TM6CK2	TM6CK1	TM6CK0	X-9
		0	0	0	0	0	0	0	0	
		Timer 6 Mode Register								
0x03F7B	TBCLR	TM6BC7	TM6BC6	TM6BC5	TM6BC4	TM6BC3	TM6BC2	TM6BC1	TM6BC0	X-7
		-	-	-	-	-	-	-	-	
		Time Base Timer Clear Control Register								
0x03F7C	TM6BEN	-	-	-	-	-	Reserved	TBEN	TM6EN	X-8
		-	-	-	-	-	0	0	0	
		Timer 6 Enable Register								
0x03F7E	TM7MD4	-	-	-	-	-	-	T7ICT2	T7CAPCLR	VIII-14
		-	-	-	-	-	-	0	0	
		Timer 7 Mode Register 4								
0x03F7F	TM8MD4	-	-	-	-	-	-	T8ICT2	T8CAPCLR	VIII-14
		-	-	-	-	-	-	0	0	
		Timer 8 Mode Register 4								
0x03F80	TM7BCL	TM7BCL7	TM7BCL6	TM7BCL5	TM7BCL4	TM7BCL3	TM7BCL2	TM7BCL1	TM7BCL0	VIII-11
		x	x	x	x	x	x	x	x	
		Timer 7 Binary Counter Lower 8 bits								
0x03F81	TM7BCH	TM7BCH7	TM7BCH6	TM7BCH5	TM7BCH4	TM7BCH3	TM7BCH2	TM7BCH1	TM7BCH0	VIII-11
		x	x	x	x	x	x	x	x	
		Timer 7 Binary Counter Upper 8 Bits								
0x03F82	TM7OC1L	TM7OC1L7	TM7OC1L6	TM7OC1L5	TM7OC1L4	TM7OC1L3	TM7OC1L2	TM7OC1L1	TM7OC1L0	VIII-9
		x	x	x	x	x	x	x	x	
		Timer 7 Compare Register 1 Lower 8bits								
0x03F83	TM7OC1H	TM7OC1H7	TM7OC1H6	TM7OC1H5	TM7OC1H4	TM7OC1H3	TM7OC1H2	TM7OC1H1	TM7OC1H0	VIII-9
		x	x	x	x	x	x	x	x	
		Timer 7 Compare Register 1 Upper 8bits								
0x03F84	TM7PR1L	TM7PR1L7	TM7PR1L6	TM7PR1L5	TM7PR1L4	TM7PR1L3	TM7PR1L2	TM7PR1L1	TM7PR1L0	VIII-9
		x	x	x	x	x	x	x	x	
		Timer 7 Preset Register 1 Lower 8bits								

Address	Register	Bit Symbol								Page
0x03F85	TM7PR1H	TM7PR1H7	TM7PR1H6	TM7PR1H5	TM7PR1H4	TM7PR1H3	TM7PR1H2	TM7PR1H1	TM7PR1H0	VIII-10
		x	x	x	x	x	x	x	x	
		Timer 7 Preset Register 1 Upper 8bits								
0x03F86	TM7ICL	TM7ICL7	TM7ICL6	TM7ICL5	TM7ICL4	TM7ICL3	TM7ICL2	TM7ICL1	TM7ICL0	VIII-11
		x	x	x	x	x	x	x	x	
		Timer 7 Input Capture Register Lower 8bits								
0x03F87	TM7ICH	TM7ICH7	TM7ICH6	TM7ICH5	TM7ICH4	TM7ICH3	TM7ICH2	TM7ICH1	TM7ICH0	VIII-11
		x	x	x	x	x	x	x	x	
		Timer 7 Input Capture Register Upper 8bits								
0x03F88	TM7MD1	Reserved	T7ICEDG1	TM7CL	TM7EN	TM7PS1	TM7PS0	TM7CK1	TM7CK0	VIII-12
		0	0	1	0	0	0	0	0	
		Timer 7 Mode Register 1								
0x03F89	TM7MD2	TM7ICEDG0	TM7PWMSL	TM7BCR	TM7PWM	TM7IRS1	TM7ICEN	TM7ICT1	TM7ICT0	VIII-13
		0	0	0	0	0	0	0	0	
		Timer7 Mode Register 2								
0x03F8A	TM7OC2L	TM7OC2L7	TM7OC2L6	TM7OC2L5	TM7OC2L4	TM7OC2L3	TM7OC2L2	TM7OC2L1	TM7OC2L0	VIII-9
		x	x	x	x	x	x	x	x	
		Timer 7 Comapre Register 2 Lower 8bits								
0x03F8B	TM7OC2H	TM7OC2H7	TM7OC2H6	TM7OC2H5	TM7OC2H4	TM7OC2H3	TM7OC2H2	TM7OC2H1	TM7OC2H0	VIII-9
		x	x	x	x	x	x	x	x	
		Timer 7 Comapre Register 2 Upper 8bits								
0x03F8C	TM7PR2L	TM7PR2L7	TM7PR2L6	TM7PR2L5	TM7PR2L4	TM7PR2L3	TM7PR2L2	TM7PR2L1	TM7PR2L0	VIII-10
		x	x	x	x	x	x	x	x	
		Timer 7 Preset Register 2 Lower 8bits								
0x03F8D	TM7PR2H	TM7PR2H7	TM7PR2H6	TM7PR2H5	TM7PR2H4	TM7PR2H3	TM7PR2H2	TM7PR2H1	TM7PR2H0	VIII-10
		x	x	x	x	x	x	x	x	
		Timer 7 Preset Register 2 Upper 8bits								
0x03F90	TM8BCL	TM8BCL7	TM8BCL6	TM8BCL5	TM8BCL4	TM8BCL3	TM8BCL2	TM8BCL1	TM8BCL0	VIII-11
		x	x	x	x	x	x	x	x	
		Timer 8 Binary Counter Lower 8 bits								
0x03F91	TM8BCH	TM8BCH7	TM8BCH6	TM8BCH5	TM8BCH4	TM8BCH3	TM8BCH2	TM8BCH1	TM8BCH0	VIII-11
		x	x	x	x	x	x	x	x	
		Timer 8 Binary Counter Upper 8 Bits								
0x03F92	TM87OC1L	TM8OC1L7	TM8OC1L6	TM8OC1L5	TM8OC1L4	TM8OC1L3	TM8OC1L2	TM8OC1L1	TM8OC1L0	VIII-9
		x	x	x	x	x	x	x	x	
		Timer 8 Compare Register 1 Lower 8bits								
0x03F93	TM8OC1H	TM8OC1H7	TM8OC1H6	TM8OC1H5	TM8OC1H4	TM8OC1H3	TM8OC1H2	TM8OC1H1	TM8OC1H0	VIII-9
		x	x	x	x	x	x	x	x	
		Timer 8 Compare Register 1 Upper 8bits								
0x03F94	TM8PR1L	TM8PR1L7	TM8PR1L6	TM8PR1L5	TM8PR1L4	TM8PR1L3	TM8PR1L2	TM8PR1L1	TM8PR1L0	VIII-9
		x	x	x	x	x	x	x	x	
		Timer 8 Preset Register 1 Lower 8bits								
0x03F95	TM8PR1H	TM8PR1H7	TM8PR1H6	TM8PR1H5	TM8PR1H4	TM8PR1H3	TM8PR1H2	TM8PR1H1	TM8PR1H0	VIII-10
		x	x	x	x	x	x	x	x	
		Timer 8 Preset Register 1 Upper 8bits								

Address	Register	Bit Symbol								Page
0x03F96	TM8ICL	TM8ICL7	TM8ICL6	TM8ICL5	TM8ICL4	TM8ICL3	TM8ICL2	TM8ICL1	TM8ICL0	VIII-11
		x	x	x	x	x	x	x	x	
		Timer 8 Input Capture Register Lower 8bits								
0x03F97	TM8ICH	TM8ICH7	TM8ICH6	TM8ICH5	TM8ICH4	TM8ICH3	TM8ICH2	TM8ICH1	TM8ICH0	VIII-11
		x	x	x	x	x	x	x	x	
		Timer 8 Input Capture Register Upper 8bits								
0x03F98	TM8MD1	Reserved	T8ICEDG1	TM8CL	TM8EN	TM8PS1	TM8PS0	TM8CK1	TM8CK0	VIII-12
		0	0	1	0	0	0	0	0	
		Timer 8 Mode Register 1								
0x03F99	TM8MD2	TM8ICEDG0	TM8PWMSL	TM8BCR	TM8PWM	TM8IRS1	TM8ICEN	TM8ICT1	TM8ICT0	VIII-13
		0	0	0	0	0	0	0	0	
		Timer 8 Mode Register 2								
0x03F9A	TM8OC2L	TM8OC2L7	TM8OC2L6	TM8OC2L5	TM8OC2L4	TM8OC2L3	TM8OC2L2	TM8OC2L1	TM8OC2L0	VIII-9
		x	x	x	x	x	x	x	x	
		Timer 8 Comapre Register 2 Lower 8bits								
0x03F9B	TM8OC2H	TM8OC2H7	TM8OC2H6	TM8OC2H5	TM8OC2H4	TM8OC2H3	TM8OC2H2	TM8OC2H1	TM8OC2H0	VIII-9
		x	x	x	x	x	x	x	x	
		Timer 8 Comapre Register 2 Upper 8bits								
0x03F9C	TM8PR2L	TM8PR2L7	TM8PR2L6	TM8PR2L5	TM8PR2L4	TM8PR2L3	TM8PR2L2	TM8PR2L1	TM8PR2L0	VIII-10
		x	x	x	x	x	x	x	x	
		Timer 8 Preset Register 2 Lower 8bits								
0x03F9D	TM8PR2H	TM8PR2H7	TM8PR2H6	TM8PR2H5	TM8PR2H4	TM8PR2H3	TM8PR2H2	TM8PR2H1	TM8PR2H0	VIII-10
		x	x	x	x	x	x	x	x	
		Timer 8 Preset Register 2 Upper 8bits								
0x03F9E	TM7MD3	TM7CKSMP	TM7BUFSEL	TM7CKEDG	-	-	-	-	-	VIII-14
		0	0	0	-	-	-	-	-	
		Timer 7 Mode Register 3								
0x03F9F	TM8MD3	TM8CKSMP	TM8BUFSEL	TM8CKEDG	T8IGBTTR	T8IGBTDT	T8IGBTEN	T8IGBT1	T8IGBT0	VIII-14
		0	0	0	0	0	0	0	0	
		Timer 8 Mode Register 3								
0x03FB0	TMCKSEL1	-	-	-	-	TM3IOSEL	TM2IOSEL	TM1IOSEL	TM0IOSEL	VI-16
		-	-	-	-	0	0	0	0	
		Timer Clock Selection Register 1								
0x03FB1	TMCKSEL2	-	-	-	-	-	TM9IOSEL	TM8IOSEL	TM7IOSEL	VIII-15
		-	-	-	-	-	0	0	0	
		Timer Clock Selection Register 2								
0x03FB2	TMINSEL1	TMINSEL17	TMINSEL16	TMINSEL15	TMINSEL14	TMINSEL13	TMINSEL12	TMINSEL11	TMINSEL10	VI-17
		0	0	0	0	0	0	0	0	
		Timer Input Selection Register 1								
0x03FB3	TMINSEL2	TMINSEL27	TMINSEL26	TMINSEL25	TMINSEL24	TMINSEL23	TMINSEL22	-	-	VIII-16
		0	0	0	0	0	0	-	-	
		Timer Input Selection Register 2								
0x03FBD	FBEWER	BEW7	BEW6	BEW5	BEW4	BEW3	BEW2	BEW1	BEW0	XVI-20
		0	0	0	0	0	0	0	0	
		Rewriting Control Register								

Address	Register	Bit Symbol								Page
0x03FBF	FEWSPD	Reserved	-	-	-	NSTOP	-	-	Reserved	III-20
		0	-	-	-	0	-	-	1	
		Internal Flash Control Register								
0x03FC5	ANCTR0	ANSH1	ANSH0	ANCK1	ANCK0	ANLADE	ANCK2	-	-	XIV-9
		0	0	0	0	0	0	-	-	
		A/D Converter Control Register 0								
0x03FC6	ANCTR1	-	-	-	-	ANCHS3	ANCHS2	ANCHS1	ANCHS0	XIV-10
		-	-	-	-	0	0	0	0	
		A/D Converter Control Register 1								
0x03FC7	ANCTR2	ANST	ANSTSEL1	ANSTSEL0	-	-	-	-	-	XIV-11
		0	0	0	-	-	-	-	-	
		A/D Converter Control Register 2								
0x03FC8	ANBUF0	ANBUF07	ANBUF06	-	-	-	-	-	-	XIV-12
		x	x	-	-	-	-	-	-	
		A/D Converter Data Storage Buffer 0								
0x03FC9	ANBUF1	ANBUF17	ANBUF16	ANBUF15	ANBUF14	ANBUF13	ANBUF12	ANBUF11	ANBUF10	XIV-12
		x	x	x	x	x	x	x	x	
		A/D Converter Data Storage Buffer 1								
0x03FD0	IRQCNT	-	-	-	P24EN	P23EN	P22EN	P21EN	P20EN	IV-44
		-	-	-	0	0	0	0	0	
		External Interrupt Pin Setting Register								
0x03FD1	NF0CTR	NF0SCK2	NF0SCK1	NF0SCK0	NF0EN1	-	-	-	-	IV-45
		0	0	0	0	-	-	-	-	
		Noise Filter 0 Control Register								
0x03FD2	NF1CTR	NF1SCK2	NF1SCK1	NF1SCK0	NF1EN1	-	-	-	-	IV-45
		0	0	0	0	-	-	-	-	
		Noise Filter 1 Control Register								
0x03FD3	NF2CTR	NF2SCK2	NF2SCK1	NF2SCK0	NF2EN1	-	-	-	-	IV-45
		0	0	0	0	-	-	-	-	
		Noise Filter 2 Control Register								
0x03FD4	NF3CTR	NF3SCK2	NF3SCK1	NF3SCK0	NF3EN1	-	-	-	-	IV-45
		0	0	0	0	-	-	-	-	
		Noise Filter 3 Control Register								
0x03FD5	NF4CTR	NF4SCK2	NF4SCK1	NF4SCK0	NF4EN1	-	-	-	-	IV-45
		0	0	0	0	-	-	-	-	
		Noise Filter 4 Control Register								
0x03FD7	LVLMD	-	-	EXLVL4	LVLEN4	EXLVL3	LVLEN3	EXLVL2	LVLEN2	IV-47
		-	-	0	0	0	0	0	0	
		External Interrupt Valid Switch Control Register								
0x03FE1	NMICR	-	-	-	-	-	IRQNPG	IRQNWDG	IRQNPRI	IV-20
		-	-	-	-	-	0	0	0	
		Non-maskable Interrupt Control Register								
0x03FE2	IRQ0ICR	IRQ0LV1	IRQ0LV0	REDG0	-	Reserved	-	IRQ0IE	IRQ0IR	IV-21
		0	0	0	-	0	-	0	0	
		External Interrupt 0 Control Register								

Address	Register	Bit Symbol								Page
0x03FE3	IRQ1ICR	IRQ1LV1	IRQ1LV0	REDG1	-	Reserved	-	IRQ1IE	IRQ1IR	IV-21
		0	0	0	-	0	-	0	0	
		External Interrupt 1 Control Register								
0x03FE4	IRQ2ICR	IRQ2LV1	IRQ2LV0	REDG2	-	Reserved	-	IRQ2IE	IRQ2IR	IV-21
		0	0	0	-	0	-	0	0	
		External Interrupt 2 Control Register								
0x03FE5	IRQ3ICR	IRQ3LV1	IRQ3LV0	REDG3	-	Reserved	-	IRQ3IE	IRQ3IR	IV-21
		0	0	0	-	0	-	0	0	
		External Interrupt 3 Control Register								
0x03FE6	IRQ4ICR	IRQ4LV1	IRQ4LV0	REDG4	-	Reserved	-	IRQ4IE	IRQ4IR	IV-21
		0	0	0	-	0	-	0	0	
		External Interrupt 4 Control Register								
0x03FE7	TS0DTICR	TS0DTLV1	TS0DTLV0	-	-	Reserved	-	TS0DTIE	TS0DTIR	IV-31
		0	0	-	-	0	-	0	0	
		Touch 0 Detect Interrupt Control Register								
0x03FE8	TS0DEICR	TS0DELV1	TS0DELV0	-	-	Reserved	-	TS0DEIE	TS0DEIR	IV-32
		0	0	-	-	0	-	0	0	
		Touch 0 Detect Error Interrupt Control Register								
0x03FE9	TS0CICR	TS0CLV1	TS0CLV0	-	-	Reserved	-	TS0CIE	TS0CIR	IV-33
		0	0	-	-	0	-	0	0	
		Touch 0 Round Interrupt Control Register								
0x03FEA	TS0ATICR	TS0ATLV1	TS0ATLV0	-	-	Reserved	-	TS0ATIE	TS0ATIR	IV-34
		0	0	-	-	0	-	0	0	
		Touch 0 Data Transmission Interrupt Control Register								
0x03FEB	TS1DTICR	TS1DTLV1	TS1DTLV0	-	-	Reserved	-	TS1DTIE	TS1DTIR	IV-31
		0	0	-	-	0	-	0	0	
		Touch 1 Detect Interrupt Control Register								
0x03FEC	TS1DEICR	TS1DELV1	TS1DELV0	-	-	Reserved	-	TS1DEIE	TS1DEIR	IV-32
		0	0	-	-	0	-	0	0	
		Touch 1 Detect Error Interrupt Control Register								
0x03FED	TS1CICR	TS1CLV1	TS1CLV0	-	-	Reserved	-	TS1CIE	TS1CIR	IV-33
		0	0	-	-	0	-	0	0	
		Touch 1 Round Interrupt Control Register								
0x03FEE	TS1ATICR	TS1ATLV1	TS1ATLV0	-	-	Reserved	-	TS1ATIE	TS1ATIR	IV-34
		0	0	-	-	0	-	0	0	
		Touch 1 Data Transmission Interrupt Control Register								
0x03FEF	TM0ICR	TM0LV1	TM0LV0	-	-	Reserved	-	TM0IE	TM0IR	IV-22
		0	0	-	-	0	-	0	0	
		Timer 0 Interrupt Control Register								
0x03FF0	TM1ICR	TM1LV1	TM1LV0	-	-	Reserved	-	TM1IE	TM1IR	IV-22
		0	0	-	-	0	-	0	0	
		Timer 1 Interrupt Control Register								
0x03FF1	TM2ICR	TM2LV1	TM2LV0	-	-	Reserved	-	TM2IE	TM2IR	IV-22
		0	0	-	-	0	-	0	0	
		Timer 2 Interrupt Control Register								

Address	Register	Bit Symbol								Page
0x03FF2	TM3ICR	TM3LV1	TM3LV0	-	-	Reserved	-	TM3IE	TM3IR	IV-22
		0	0	-	-	0	-	0	0	
		Timer 3 Interrupt Control Register								
0x03FF3	TM6ICR	TM6LV1	TM6LV0	-	-	Reserved	-	TM6IE	TM6IR	IV-22
		0	0	-	-	0	-	0	0	
		Timer 6 Interrupt Control Register								
0x03FF4	TBICR	TBLV1	TBLV0	-	-	Reserved	-	TBIE	TBIR	IV-23
		0	0	-	-	0	-	0	0	
		Time Base Interrupt Control Register								
0x03FF5	TM7ICR	TM7LV1	TM7LV0	-	-	Reserved	-	TM7IE	TM7IR	IV-24
		0	0	-	-	0	-	0	0	
		Timer 7 Interrupt Control Register								
0x03FF6	TM7OC2ICR	TM7OC2LV1	TM7OC2LV0	-	-	Reserved	-	TM7OC2IE	TM7OC2IR	IV-25
		0	0	-	-	0	-	0	0	
		Timer 7 Compare 2-Match Interrupt Control Register								
0x03FF7	TM8ICR	TM8LV1	TM8LV0	-	-	Reserved	-	TM8IE	TM8IR	IV-24
		0	0	-	-	0	-	0	0	
		Timer 8 Interrupt Control Register								
0x03FF8	TM8OC2ICR	TM8OC2LV1	TM8OC2LV0	-	-	Reserved	-	TM8OC2IE	TM8OC2IR	IV-25
		0	0	-	-	0	-	0	0	
		Timer 8 Compare 2-Match Interrupt Control Register								
0x03FF9	PWMOVICR	PWMOVLV1	PWMOVLV0	-	-	Reserved	-	PWMOVIE	PWMOVIR	IV-26
		0	0	-	-	0	-	0	0	
		Timer 9 Overflow Interrupt Control Register								
0x03FFA	PWMUDICR	PWMUDLV1	PWMUDLV0	-	-	Reserved	-	PWMUDIE	PWMUDIR	IV-27
		0	0	-	-	0	-	0	0	
		Timer 9 Underflow Interrupt Control Register								
0x03FFB	SC0TICR	SC0TLV1	SC0TLV0	-	-	Reserved	-	SC0TIE	SC0TIR	IV-28
		0	0	-	-	0	-	0	0	
		Serial 0 Transmission Interrupt Control Register								
0x03FFC	SC0RICR	SC0RLV1	SC0RLV0	-	-	Reserved	-	SC0RIE	SC0RIR	IV-29
		0	0	-	-	0	-	0	0	
		Serial 0 UART Reception Interrupt Control Register								
0x03FFD	ADICR	ADLV1	ADLV0	-	-	Reserved	-	ADIE	ADIR	IV-30
		0	0	-	-	0	-	0	0	
		A/D Conversion Interrupt Control Register								
0x03FFE	PERIILR	PERILV1	PERILV0	-	-	Reserved	-	-	-	IV-35
		0	0	-	-	0	-	-	-	
		Peripheral Function Group Interrupt Level Setting Register								

17.2 Instruction Set

MN101E SERIES INSTRUCTION SET

Data Move Instructions

Group	Mnemonic	Operation	Flag				Code Size	Cycle	Re-peat	Machine Code												Notes
			VF	NF	CF	ZF				Ext.	1	2	3	4	5	6	7	8	9	10	11	
MOV	MOV Dn,Dm	Dn→Dm	--	--	--	--	2	1			1010	DnDm										
	MOV imm8,Dm	imm8→Dm	--	--	--	--	4	2			1010	DmDm	<#8.	...>								
	MOV Dn,PSW	Dn→PSW	●	●	●	●	3	3		0010	1001	01Dn										
	MOV PSW,Dm	PSW→Dm	--	--	--	--	3	2		0010	0001	01Dm										
	MOV (An),Dm	mem8(An)→Dm	--	--	--	--	2	2			0100	1ADm										
	MOV (d8,An),Dm	mem8(d8+An)→Dm	--	--	--	--	4	2			0110	1ADm	<d8.	...>								*1
	MOV (d16,An),Dm	mem8(d16+An)→Dm	--	--	--	--	7	4		0010	0110	1ADm	<d16	....	....	...>						
	MOV (d4,SP),Dm	mem8(d4+SP)→Dm	--	--	--	--	3	2			0110	01Dm	<d4>									*2
	MOV (d8,SP),Dm	mem8(d8+SP)→Dm	--	--	--	--	5	3		0010	0110	01Dm	<d8.	...>								*3
	MOV (d16,SP),Dm	mem8(d16+SP)→Dm	--	--	--	--	7	4		0010	0110	00Dm	<d16	....	....	...>						
	MOV (io8),Dm	mem8(IOTOP+io8)→Dm	--	--	--	--	4	2			0110	00Dm	<io8	...>								
	MOV (abs8),Dm	mem8(abs8)→Dm	--	--	--	--	4	2			0100	01Dm	<abs	8..>								
	MOV (abs12),Dm	mem8(abs12)→Dm	--	--	--	--	5	2			0100	00Dm	<abs	12..	...>							
	MOV (abs16),Dm	mem8(abs16)→Dm	--	--	--	--	7	4		0010	1100	00Dm	<abs	16..	....	...>						
	MOV Dn,(Am)	Dn→mem8(Am)	--	--	--	--	2	2			0101	1aDn										
	MOV Dn,(d8,Am)	Dn→mem8(d8+Am)	--	--	--	--	4	2			0111	1aDn	<d8.	...>								*1
	MOV Dn,(d16,Am)	Dn→mem8(d16+Am)	--	--	--	--	7	4		0010	0111	1aDn	<d16	....	....	...>						
	MOV Dn,(d4,SP)	Dn→mem8(d4+SP)	--	--	--	--	3	2			0111	01Dn	<d4>									*2
	MOV Dn,(d8,SP)	Dn→mem8(d8+SP)	--	--	--	--	5	3		0010	0111	01Dn	<d8.	...>								*3
	MOV Dn,(d16,SP)	Dn→mem8(d16+SP)	--	--	--	--	7	4		0010	0111	00Dn	<d16	....	....	...>						
	MOV Dn,(io8)	Dn→mem8(IOTOP+io8)	--	--	--	--	4	2			0111	00Dn	<io8	...>								
	MOV Dn,(abs8)	Dn→mem8(abs8)	--	--	--	--	4	2			0101	01Dn	<abs	8..>								
	MOV Dn,(abs12)	Dn→mem8(abs12)	--	--	--	--	5	2			0101	00Dn	<abs	12..	...>							
	MOV Dn,(abs16)	Dn→mem8(abs16)	--	--	--	--	7	4		0010	1101	00Dn	<abs	16..	....	...>						
	MOV imm8,(io8)	imm8→mem8(IOTOP+io8)	--	--	--	--	6	3			0000	0010	<io8	...>	<#8.	...>						
	MOV imm8,(abs8)	imm8→mem8(abs8)	--	--	--	--	6	3			0001	0100	<abs	8..>	<#8.	...>						
	MOV imm8,(abs12)	imm8→mem8(abs12)	--	--	--	--	7	3			0001	0101	<abs	12..	...>	<#8.	...>					
	MOV imm8,(abs16)	imm8→mem8(abs16)	--	--	--	--	9	5		0011	1101	1001	<abs	16..	....	...>	<#8.	...>				
	MOV Dn,(HA)	Dn→mem8(HA)	--	--	--	--	2	2			1101	00Dn										
MOVW	MOVW (An),DWm	mem16(An)→DWm	--	--	--	--	2	3			1110	00Ad										
	MOVW (An),Am	mem16(An)→Am	--	--	--	--	3	4		0010	1110	10Aa										*4
	MOVW (d4,SP),DWm	mem16(d4+SP)→DWm	--	--	--	--	3	3			1110	011d	<d4>									*2
	MOVW (d4,SP),Am	mem16(d4+SP)→Am	--	--	--	--	3	3			1110	010a	<d4>									*2
	MOVW (d8,SP),DWm	mem16(d8+SP)→DWm	--	--	--	--	5	4		0010	1110	011d	<d8.	...>								*3
	MOVW (d8,SP),Am	mem16(d8+SP)→Am	--	--	--	--	5	4		0010	1110	010a	<d8.	...>								*3
	MOVW (d16,SP),DWm	mem16(d16+SP)→DWm	--	--	--	--	7	5		0010	1110	001d	<d16	....	....	...>						
	MOVW (d16,SP),Am	mem16(d16+SP)→Am	--	--	--	--	7	5		0010	1110	000a	<d16	....	....	...>						
	MOVW (abs8),DWm	mem16(abs8)→DWm	--	--	--	--	4	3			1100	011d	<abs	8..>								
	MOVW (abs8),Am	mem16(abs8)→Am	--	--	--	--	4	3			1100	010a	<abs	8..>								
	MOVW (abs16),DWm	mem16(abs16)→DWm	--	--	--	--	7	5		0010	1100	011d	<abs	16..	....	...>						
	MOVW (abs16),Am	mem16(abs16)→Am	--	--	--	--	7	5		0010	1100	010a	<abs	16..	....	...>						
	MOVW DWn,(Am)	DWn→mem16(Am)	--	--	--	--	2	3			1111	00aD										
	MOVW An,(Am)	An→mem16(Am)	--	--	--	--	3	4		0010	1111	10aA										*4
	MOVW DWn,(d4,SP)	DWn→mem16(d4+SP)	--	--	--	--	3	3			1111	011D	<d4>									*2
	MOVW An,(d4,SP)	An→mem16(d4+SP)	--	--	--	--	3	3			1111	010A	<d4>									*2
	MOVW DWn,(d8,SP)	DWn→mem16(d8+SP)	--	--	--	--	5	4		0010	1111	011D	<d8.	...>								*3
	MOVW An,(d8,SP)	An→mem16(d8+SP)	--	--	--	--	5	4		0010	1111	010A	<d8.	...>								*3
	MOVW DWn,(d16,SP)	DWn→mem16(d16+SP)	--	--	--	--	7	5		0010	1111	001D	<d16	....	....	...>						
	MOVW An,(d16,SP)	An→mem16(d16+SP)	--	--	--	--	7	5		0010	1111	000A	<d16	....	....	...>						
	MOVW DWn,(abs8)	DWn→mem16(abs8)	--	--	--	--	4	3			1101	011D	<abs	8..>								
	MOVW An,(abs8)	An→mem16(abs8)	--	--	--	--	4	3			1101	010A	<abs	8..>								
	MOVW DWn,(abs16)	DWn→mem16(abs16)	--	--	--	--	7	5		0010	1101	011D	<abs	16..	....	...>						
	MOVW An,(abs16)	An→mem16(abs16)	--	--	--	--	7	5		0010	1101	010A	<abs	16..	....	...>						
	MOVW DWn,(HA)	DWn→mem16(HA)	--	--	--	--	2	3			1001	010D										
	MOVW An,(HA)	An→mem16(HA)	--	--	--	--	2	3			1001	011A										
	MOVW imm8,DWm	sign(imm8)→DWm	--	--	--	--	4	2			0000	110d	<#8.	...>								*5
	MOVW imm8,Am	zero(imm8)→Am	--	--	--	--	4	2			0000	111a	<#8.	...>								*6
	MOVW imm16,DWm	imm16→DWm	--	--	--	--	6	3			1100	111d	<#16	....	....	...>						

*1 d8 sign-extension
*2 d4 zero-extension
*3 d8 zero-extension
*4 A=An, a=Am
*5 #8 sign-extension
*6 #8 zero-extension

MN101E SERIES INSTRUCTION SET

Group	Mnemonic	Operation	Flag				Code Size	Cycle	Re-peat	exten-sion	Machine Code											Notes
			VF	NF	CF	ZF					1	2	3	4	5	6	7	8	9	10	11	
	MOVW imm16,Am	imm16→Am	--	--	--	--	6	3			1101	111a	<#16	....	....	...>						
	MOVW SP,Am	SP→Am	--	--	--	--	3	3		0010	0000	100a										
	MOVW An,SP	An→SP	--	--	--	--	3	3		0010	0000	101A										
	MOVW DWn,DWm	DWn→DWm	--	--	--	--	3	3		0010	1000	00Dd										*1
	MOVW DWn,Am	DWn→Am	--	--	--	--	3	3		0010	0100	11Da										
	MOVW An,DWm	An→DWm	--	--	--	--	3	3		0010	1100	11Ad										
	MOVW An,Am	An→Am	--	--	--	--	3	3		0010	0000	00Aa										*2
PUSH	PUSH Dn	SP-1→SP,Dn→mem8(SP)	--	--	--	--	2	3			1111	10Dn										
	PUSH An	SP-2→SP,An→mem16(SP)	--	--	--	--	2	5			0001	011A										
POP	POP Dn	mem8(SP)→Dn,SP+1→SP	--	--	--	--	2	3			1110	10Dn										
	POP An	mem16(SP)→An,SP+2→SP	--	--	--	--	2	4			0000	011A										
EXT	EXT Dn,DWm	sign(Dn)→DWm	--	--	--	--	3	3		0010	1001	000d										*3

Arithmetic manupulation instructions

ADD	ADD Dn,Dm	Dm+Dn→Dm	●	●	●	●	3	2		0011	0011	DnDm										
	ADD imm4,Dm	Dm+sign(imm4)→Dm	●	●	●	●	3	2			1000	00Dm	<#4>									*6
	ADD imm8,Dm	Dm+imm8→Dm	●	●	●	●	4	2			0000	10Dm	<#8.	...>								
ADDC	ADDC Dn,Dm	Dm+Dn+CF→Dm	●	●	●	●	3	2	○	0011	1011	DnDm										
ADDW	ADDW DWn,DWm	DWm+DWn→DWm	●	●	●	●	3	3	○	0010	0101	00Dd										*1
	ADDW DWn,Am	Am+DWn→Am	●	●	●	●	3	3	○	0010	0101	10Da										
	ADDW imm4,Am	Am+sign(imm4)→Am	●	●	●	●	3	2			1110	110a	<#4>									*6
	ADDW imm8,Am	Am+sign(imm8)→Am	●	●	●	●	5	3		0010	1110	110a	<#8.	...>								*7
	ADDW imm16,Am	Am+imm16→Am	●	●	●	●	7	4		0010	0101	011a	<#16	....	....	...>						
	ADDW imm4,SP	SP+sign(imm4)→SP	--	--	--	--	3	2			1111	1101	<#4>									*6
	ADDW imm8,SP	SP+sign(imm8)→SP	--	--	--	--	4	2			1111	1100	<#8.	...>								*7
	ADDW imm16,SP	SP+imm16→SP	--	--	--	--	7	4		0010	1111	1100	<#16	....	....	...>						
	ADDW imm16,DWm	DWm+imm16→DWm	●	●	●	●	7	4		0010	0101	010d	<#16	....	....	...>						
ADDUW	ADDUW Dn,Am	Am+zero(Dn)→Am	●	●	●	●	3	3	○	0010	1000	1aDn										*8
ADDSW	ADDSW Dn,Am	Am+sign(Dn)→Am	●	●	●	●	3	3	○	0010	1001	1aDn										
SUB	SUB Dn,Dm(when Dn≠Dm)	Dm-Dn→Dm	●	●	●	●	3	2	○	0010	1010	DnDm										
	SUB Dn,Dn	Dn-Dn→Dn	0	0	0	1	2	1			1000	01Dn										
	SUB imm8,Dm	Dm-imm8→Dm	●	●	●	●	5	3		0010	1010	DmDm	<#8.	...>								
SUBC	SUBC Dn,Dm	Dm-Dn-CF→Dm	●	●	●	●	3	2	○	0010	1011	DnDm										
SUBW	SUBW DWn,DWm	DWm-DWn→DWm	●	●	●	●	3	3		0010	0100	00Dd										*1
	SUBW DWn,Am	Am-DWn→Am	●	●	●	●	3	3		0010	0100	10Da										
	SUBW imm16,DWm	DWm-imm16→DWm	●	●	●	●	7	4		0010	0100	010d	<#16	....	....	...>						
	SUBW imm16,Am	Am-imm16→Am	●	●	●	●	7	4		0010	0100	011a	<#16	....	....	...>						
MULU	MULU Dn,Dm	Dm*Dn→DWk	0	●	●	●	3	8		0010	1111	111D										*4
DIVU	DIVU Dn,DWm	DWm/Dn→DWm-I...DWm-h	●	●	●	●	3	9		0010	1110	111d										*5
CMP	CMP Dn,Dm	Dm-Dn...PSW	●	●	●	●	3	2		0011	0010	DnDm										
	CMP imm8,Dm	Dm-imm8...PSW	●	●	●	●	4	2			1100	00Dm	<#8.	...>								
	CMP imm8,(abs8)	mem8(abs8)-imm8...PSW	●	●	●	●	6	3			0000	0100	<abs	8..>	<#8.	...>						
	CMP imm8,(abs12)	mem8(abs12)-imm8...PSW	●	●	●	●	7	3			0000	0101	<abs	12..	...>	<#8.	...>					
	CMP imm8,(abs16)	mem8(abs16)-imm8...PSW	●	●	●	●	9	5		0011	1101	1000	<abs	16..	....	...>	<#8.	...>				
CMPW	CMPW DWn,DWm	DWm-DWn...PSW	●	●	●	●	3	3		0010	1000	01Dd										*1
	CMPW DWn,Am	Am-DWn...PSW	●	●	●	●	3	3		0010	0101	11Da										
	CMPW An,Am	Am-An...PSW	●	●	●	●	3	3		0010	0000	01Aa										*2
	CMPW imm16,DWm	DWm-imm16...PSW	●	●	●	●	6	3			1100	110d	<#16	....	....	...>						
	CMPW imm16,Am	Am-imm16...PSW	●	●	●	●	6	3			1101	110a	<#16	....	....	...>						

Logical manipulation instructions

AND	AND Dn,Dm	Dm&Dn→Dm	0	●	0	●	3	2		0011	0111	DnDm										
	AND imm8,Dm	Dm&imm8→Dm	0	●	0	●	4	2			0001	11Dm	<#8.	...>								
	AND imm8,PSW	PSW&imm8→PSW	●	●	●	●	5	3		0010	1001	0010	<#8.	...>								
OR	OR Dn,Dm	DmIDn→Dm	0	●	0	●	3	2		0011	0110	DnDm										
	OR imm8,Dm	DmIimm8→Dm	0	●	0	●	4	2			0001	10Dm	<#8.	...>								
	OR imm8,PSW	PSWIimm8→PSW	●	●	●	●	5	3		0010	1001	0011	<#8.	...>								
XOR	XOR Dn,Dm	Dm^Dn→Dm	0	●	0	●	3	2		0011	1010	DnDm										*9
	XOR imm8,Dm	Dm^imm8→Dm	0	●	0	●	5	3		0011	1010	DmDm	<#8.	...>								

*1	D=DWn, d=DWm	*5	D=DWm	*9	m≠n
*2	A=An, a=Am	*6	#4 sign-extension		
*3	d=DWm	*7	#8 sign-extension		
*4	D=DWk	*8	Dn zero extension		

MN101E SERIES INSTRUCTION SET

Group	Mnemonic	Operation	Flag				Code Size	Cycle	Re-peat	Exten sion	Machine Code											Notes
			VF	NF	CF	ZF					1	2	3	4	5	6	7	8	9	10	11	
NOT	NOT Dn	¯Dn→Dn=	0	●	0	●	3	2		0010	0010	10Dn										
ASR	ASR Dn	Dn.msb→temp,Dn.lsb→CF Dn>>1→Dn,temp→Dn.msb	0	--	●	●	3	2	○	0010	0011	10Dn										
LSR	LSR Dn	Dn.lsb→CF,Dn>>1→Dn 0→Dn.msb	0	0	●	●	3	2	○	0010	0011	11Dn										
ROR	ROR Dn	Dn.lsb→temp,Dn>>1→Dn CF→Dn.msb,temp→CF	0	●	●	●	3	2	○	0010	0010	11Dn										

Bit manipulation instructions

BSET	BSET (io8)bp	mem8(IOTOP+io8)&bpdata...PSW 1→mem8(IOTOP+io8)bp	0	●	0	●	5	5		0011	1000	0bp.	<io8	...>								
	BSET (abs8)bp	mem8(abs8)&bpdata...PSW 1→mem8(abs8)bp	0	●	0	●	4	4			1011	0bp.	<abs	8..>								
	BSET (abs16)bp	mem8(abs16)&bpdata...PSW 1→mem8(abs16)bp	0	●	0	●	7	6		0011	1100	0bp.	<abs	16..	....	...>						
BCLR	BCLR (io8)bp	mem8(IOTOP+io8)&bpdata...PSW 0→mem8(IOTOP+io8)bp	0	●	0	●	5	5		0011	1000	1bp.	<io8	...>								
	BCLR (abs8)bp	mem8(abs8)&bpdata...PSW 0→mem8(abs8)bp	0	●	0	●	4	4			1011	1bp.	<abs	8..>								
	BCLR (abs16)bp	mem8(abs16)&bpdata...PSW 0→mem8(abs16)bp	0	●	0	●	7	6		0011	1100	1bp.	<abs	16..	....	...>						
BTST	BTST imm8,Dm	Dm&imm8...PSW	0	●	0	●	5	3		0010	0000	11Dm	<#8.	...>								
	BTST (abs16)bp	mem8(abs16)&bpdata...PSW	0	●	0	●	7	5		0011	1101	0bp.	<abs	16..	....	...>						

Branch instructions

Bcc	BEQ label	if(ZF=1), PC+3+d4(label)+H→PC if(ZF=0), PC+3→PC	--	--	--	--	3	2/3			1001	000H	<d4>									*1
	BEQ label	if(ZF=1), PC+4+d7(label)+H→PC if(ZF=0), PC+4→PC	--	--	--	--	4	2/3			1000	1010	<d7.	...H								*2
	BEQ label	if(ZF=1), PC+5+d11(label)+H→PC if(ZF=0), PC+5→PC	--	--	--	--	5	2/3			1001	1010	<d11	....	...H							*3
	BNE label	if(ZF=0), PC+3+d4(label)+H→PC if(ZF=1), PC+3→PC	--	--	--	--	3	2/3			1001	001H	<d4>									1
	BNE label	if(ZF=0), PC+4+d7(label)+H→PC if(ZF=1), PC+4→PC	--	--	--	--	4	2/3			1000	1011	<d7.	...H								*2
	BNE label	if(ZF=0), PC+5+d11(label)+H→PC if(ZF=1), PC+5→PC	--	--	--	--	5	2/3			1001	1011	<d11	....	...H							*3
	BGE label	if((VF^NF)=0),PC+4+d7(label)+H→PC if((VF^NF)=1),PC+4→PC	--	--	--	--	4	2/3			1000	1000	<d7.	...H								*2
	BGE label	if((VF^NF)=0),PC+5+d11(label)+H→PC if((VF^NF)=1),PC+5→PC	--	--	--	--	5	2/3			1001	1000	<d11	....	...H							*3
	BCC label	if(CF=0),PC+4+d7(label)+H→PC if(CF=1), PC+4→PC	--	--	--	--	4	2/3			1000	1100	<d7.	...H								*2
	BCC label	if(CF=0), PC+5+d11(label)+H→PC if(CF=1), PC+5→PC	--	--	--	--	5	2/3			1001	1100	<d11	....	...H							*3
	BCS label	if(CF=1),PC+4+d7(label)+H→PC if(CF=0), PC+4→PC	--	--	--	--	4	2/3			1000	1101	<d7.	...H								*2
	BCS label	if(CF=1), PC+5+d11(label)+H→PC if(CF=0), PC+5→PC	--	--	--	--	5	2/3			1001	1101	<d11	....	...H							*3
	BLT label	if((VF^NF)=1),PC+4+d7(label)+H→PC if((VF^NF)=0),PC+4→PC	--	--	--	--	4	2/3			1000	1110	<d7.	...H								*2
	BLT label	if((VF^NF)=1),PC+5+d11(label)+H→PC if((VF^NF)=0),PC+5→PC	--	--	--	--	5	2/3			1001	1110	<d11	....	...H							*3
	BLE label	if((VF^NF)|ZF=1),PC+4+d7(label)+H→PC if((VF^NF)|ZF=0),PC+4→PC	--	--	--	--	4	2/3			1000	1111	<d7.	...H								*2
	BLE label	if((VF^NF)|ZF=1),PC+5+d11(label)+H→PC if((VF^NF)|ZF=0),PC+5→PC	--	--	--	--	5	2/3			1001	1111	<d11	....	...H							*3
	BGT label	if((VF^NF)|ZF=0),PC+5+d7(label)+H→PC if((VF^NF)|ZF=1),PC+5→PC	--	--	--	--	5	3/4		0010	0010	0001	<d7.	...H								*2

*1 d4 sign-extension
*2 d7 sign-extension
*3 d11 sign-extension

MN101E SERIES INSTRUCTION SET

Group	Mnemonic	Operation	Flag				Code Size	Cycle	Re-peat	Exten-sion	Machine Code											Notes
			VF	NF	CF	ZF					1	2	3	4	5	6	7	8	9	10	11	
Bcc	BGT label	if((VF^NF)|ZF=0),PC+6+d11(label)+H→PC if((VF^NF)|ZF=1),PC+6→PC	--	--	--	--	6	3/4		0010	0011	0001	<d11	....	...H							*3
	BHI label	if(CFIZF=0),PC+5+d7(label)+H→PC if(CFIZF=1), PC+5→PC	--	--	--	--	5	3/4		0010	0010	0010	<d7.	...H								*2
	BHI label	if(CFIZF=0),PC+6+d11(label)+H→PC if(CFIZF=1), PC+6→PC	--	--	--	--	6	3/4		0010	0011	0010	<d11	....	...H							*3
	BLS label	if(CFIZF=1),PC+5+d7(label)+H→PC if(CFIZF=0), PC+5→PC	--	--	--	--	5	3/4		0010	0010	0011	<d7.	...H								*2
	BLS label	if(CFIZF=1),PC+6+d11(label)+H→PC if(CFIZF=0), PC+6→PC	--	--	--	--	6	3/4		0010	0011	0011	<d11	....	...H							*3
	BNC label	if(NF=0),PC+5+d7(label)+H→PC if(NF=1),PC+5→PC	--	--	--	--	5	3/4		0010	0010	0100	<d7.	...H								*2
	BNC label	if(NF=0),PC+6+d11(label)+H→PC if(NF=1),PC+6→PC	--	--	--	--	6	3/4		0010	0011	0100	<d11	....	...H							*3
	BNS label	if(NF=1),PC+5+d7(label)+H→PC if(NF=0),PC+5→PC	--	--	--	--	5	3/4		0010	0010	0101	<d7.	...H								*2
	BNS label	if(NF=1),PC+6+d11(label)+H→PC if(NF=0),PC+6→PC	--	--	--	--	6	3/4		0010	0011	0101	<d11	....	...H							*3
	BVC label	if(VF=0),PC+5+d7(label)+H→PC if(VF=1),PC+5→PC	--	--	--	--	5	3/4		0010	0010	0110	<d7.	...H								*2
	BVC label	if(VF=0),PC+6+d11(label)+H→PC if(VF=1),PC+6→PC	--	--	--	--	6	3/4		0010	0011	0110	<d11	....	...H							*3
	BVS label	if(VF=1),PC+5+d7(label)+H→PC if(VF=0),PC+5→PC	--	--	--	--	5	3/4		0010	0010	0111	<d7.	...H								*2
	BVS label	if(VF=1),PC+6+d11(label)+H→PC if(VF=0),PC+6→PC	--	--	--	--	6	3/4		0010	0011	0111	<d11	....	...H							*3
	BRA label	PC+3+d4(label)+H→PC	--	--	--	--	3	3			1110	111H	<d4>									*1
	BRA label	PC+4+d7(label)+H→PC	--	--	--	--	4	3			1000	1001	<d7.	...H								*2
	BRA label	PC+5+d11(label)+H→PC	--	--	--	--	5	3			1001	1001	<d11	....	...H							*3
CBEQ	CBEQ imm8,Dm,label	if(Dm=imm8),PC+6+d7(label)+H→PC if(Dm≠imm8),PC+6→PC	●	●	●	●	6	3/4			1100	10Dm	<#8.	...>	<d7.	...H						*2
	CBEQ imm8,Dm,label	if(Dm=imm8),PC+8+d11(label)+H→PC if(Dm≠imm8),PC+8→PC	●	●	●	●	8	4/5		0010	1100	10Dm	<#8.	...>	<d11	....	...H					*3
	CBEQ imm8,(abs8),label	if(mem8(abs8)=imm8),PC+9+d7(label)+H→PC if(mem8(abs8)≠imm8),PC+9→PC	●	●	●	●	9	6/7		0010	1101	1100	<abs	8..>	<#8.	...>	<d7.	...H				*2
	CBEQ imm8,(abs8),label	if(mem8(abs8)=imm8),PC+10+d11(label)+H→PC if(mem8(abs8)≠imm8),PC+10→PC	●	●	●	●	10	6/7		0010	1101	1101	<abs	8..>	<#8.	...>	<d11	....	...H			*3
	CBEQ imm8,(abs16),label	if(mem8(abs16)=imm8),PC+11+d7(label)+H→PC if(mem8(abs16)≠imm8),PC+11→PC	●	●	●	●	11	7/8		0011	1101	1100	<abs	16..	....	...>	<#8.	...>	<d7.	...H		*2
	CBEQ imm8,(abs16),label	if(mem8(abs16)=imm8),PC+12+d11(label)+H→PC if(mem8(abs16)≠imm8),PC+12→PC	●	●	●	●	12	7/8		0011	1101	1101	<abs	16..	....	...>	<#8.	...>	<d11	....	...H	*3
CBNE	CBNE imm8,Dm,label	if(Dm≠imm8),PC+6+d7(label)+H→PC if(Dm=imm8),PC+6→PC	●	●	●	●	6	3/4			1101	10Dm	<#8.	...>	<d7.	..H>						*2
	CBNE imm8,Dm,label	if(Dm≠imm8),PC+8+d11(label)+H→PC if(Dm=imm8),PC+8→PC	●	●	●	●	8	4/5		0010	1101	10Dm	<#8.	...>	<d11	....	...H					*3
	CBNE imm8,(abs8),label	if(mem8(abs8)≠imm8),PC+9+d7(label)+H→PC if(mem8(abs8)=imm8),PC+9→PC	●	●	●	●	9	6/7		0010	1101	1110	<abs	8..>	<#8.	...>	<d7.	...H				*2
	CBNE imm8,(abs8),label	if(mem8(abs8)≠imm8),PC+10+d11(label)+H→PC if(mem8(abs8)=imm8),PC+10→PC	●	●	●	●	10	6/7		0010	1101	1111	<abs	8..>	<#8.	...>	<d11	....	...H			*3
	CBNE imm8,(abs16),label	if(mem8(abs16)≠imm8),PC+11+d7(label)+H→PC if(mem8(abs16)=imm8),PC+11→PC	●	●	●	●	11	7/8		0011	1101	1110	<abs	16..	....	...>	<#8.	...>	<d7.	...H		*2
	CBNE imm8,(abs16),label	if(mem8(abs16)≠imm8),PC+12+d11(label)+H→PC if(mem8(abs16)=imm8),PC+12→PC	●	●	●	●	12	7/8		0011	1101	1111	<abs	16..	....	...>	<#8.	...>	<d11	....	...H	*3
TBZ	TBZ (abs8)bp,label	if(mem8(abs8)bp=0),PC+7+d7(label)+H→PC if(mem8(abs8)bp=1),PC+7→PC	0	●	0	●	7	6/7		0011	0000	0bp.	<abs	8..>	<d7.	...H						*2
	TBZ (abs8)bp,label	if(mem8(abs8)bp=0),PC+8+d11(label)+H→PC if(mem8(abs8)bp=1),PC+8→PC	0	●	0	●	8	6/7		0011	0000	1bp.	<abs	8..>	<d11	....	...H					*3

*1 d4 sign-extension
*2 d7 sign-extension
*3 d11 sign-extension

MN101E SERIES INSTRUCTION SET

Group	Mnemonic	Operation	Flag				Code Size	Cycle	Re-peat	Exten-sion	Machine Code											Notes
			VF	NF	CF	ZF					1	2	3	4	5	6	7	8	9	10	11	
TBZ	TBZ (io8)bp,label	if(mem8(IOTOP+io8)bp=0),PC+7+d7(label)+H→PC if(mem8(IOTOP+io8)bp=1),PC+7→PC	0	●	0	●	7	6/7		0011	0100	0bp.	<io8	...>	<d7.	...H						*1
	TBZ (io8)bp,label	if(mem8(IOTOP+io8)bp=0),PC+8+d11(label)+H→PC if(mem8(IOTOP+io8)bp=1),PC+8→PC	0	●	0	●	8	6/7		0011	0100	1bp.	<io8	...>	<d11	....	...H					*2
	TBZ (abs16)bp,label	if(mem8(abs16)bp=0),PC+9+d7(label)+H→PC if(mem8(abs16)bp=1),PC+9→PC	0	●	0	●	9	7/8		0011	1110	0bp.	<abs	16..	....	...>	<d7.	...H				*1
	TBZ (abs16)bp,label	if(mem8(abs16)bp=0),PC+10+d11(label)+H→PC if(mem8(abs16)bp=1),PC+10→PC	0	●	0	●	10	7/8		0011	1110	1bp.	<abs	16..	....	...>	<d11	....	...H			*2
TBNZ	TBNZ (abs8)bp,label	if(mem8(abs8)bp=1),PC+7+d7(label)+H→PC if(mem8(abs8)bp=0),PC+7→PC	0	●	0	●	7	6/7		0011	0001	0bp.	<abs	8..>	<d7.	...H						*1
	TBNZ (abs8)bp,label	if(mem8(abs8)bp=1),PC+8+d11(label)+H→PC if(mem8(abs8)bp=0),PC+8→PC	0	●	0	●	8	6/7		0011	0001	1bp.	<abs	8..>	<d11	....	...H					*2
	TBNZ (io8)bp,label	if(mem8(io)bp=1),PC+7+d7(label)+H→PC if(mem8(io)bp=0),PC+7→PC	0	●	0	●	7	6/7		0011	0101	0bp.	<io8	...>	<d7.	...H						*1
	TBNZ (io8)bp,label	if(mem8(io)bp=1),PC+8+d11(label)+H→PC if(mem8(io)bp=0),PC+8→PC	0	●	0	●	8	6/7		0011	0101	1bp.	<io8	...>	<d11	....	...H					*2
	TBNZ (abs16)bp,label	if(mem8(abs16)bp=1),PC+9+d7(label)+H→PC if(mem8(abs16)bp=0),PC+9→PC	0	●	0	●	9	7/8		0011	1111	0bp.	<abs	16..	....	...>	<d7.	...H				*1
	TBNZ (abs16)bp,label	if(mem8(abs16)bp=1),PC+10+d11(label)+H→PC if(mem8(abs16)bp=0),PC+10→PC	0	●	0	●	10	7/8		0011	1111	1bp.	<abs	16..	....	...>	<d11	....	...H			*2
JMP	JMP (An)	0→PC.17-16,An→PC.15-0,0→PC.H	---	---	---	---	3	4		0010	0001	00A0										
	JMP label	abs18(label)+H→PC	---	---	---	---	7	5		0011	1001	0aaH	<abs	18.b	p15~	0..>						*5
	JMP label	abs20(label)+H→PC	---	---	---	---	9	6		0011	1101	1010	000B	bbbH	<abs	20.b	p15~	0..>				*6*7
JSR	JSR (An)	SP-3→SP,(PC+3).bp7-0→mem8(SP) (PC+3).bp15-8→mem8(SP+1) (PC+3).H→mem8(SP+2).bp7, 0→mem8(SP+2).bp6-4, (PC+3).bp19-16→mem8(SP+2).bp3-0 0→PC.bp19-16 An→PC.bp15-0,0→PC.H	---	---	---	---	3	7		0010	0001	00A1										
	JSR label	SP-3→SP,(PC+5).bp7-0→mem8(SP) (PC+5).bp15-8→mem8(SP+1) (PC+5).H→mem8(SP+2).bp7, 0→mem8(SP+2).bp6-4, (PC+5).bp19-16→mem8(SP+2).bp3-0 PC+5+d12(label)+H→PC	---	---	---	---	5	6			0001	000H	<d12	....	...>							*3
	JSR label	SP-3→SP,(PC+6).bp7-0→mem8(SP) (PC+6).bp15-8→mem8(SP+1) (PC+6).H→mem8(SP+2).bp7, 0→mem8(SP+2).bp6-4, (PC+6).bp19-16→mem8(SP+2).bp3-0 PC+6+d16(label)+H→PC	---	---	---	---	6	7			0001	001H	<d16	....	....	...>						*4
	JSR label	SP-3→SP,(PC+7).bp7-0→mem8(SP) (PC+7).bp15-8→mem8(SP+1) (PC+7).H→mem8(SP+2).bp7, 0→mem8(SP+2).bp6-4, (PC+7).bp19-16→mem8(SP+2).bp3-0 abs18(label)+H→PC	---	---	---	---	7	8		0011	1001	1aaH	<abs	18.b	p15~	0..>						*5
	JSR label	SP-3→SP,(PC+7).bp7-0→mem8(SP) (PC+7).bp15-8→mem8(SP+1) (PC+7).H→mem8(SP+2).bp7, 0→mem8(SP+2).bp6-4, (PC+7).bp19-16→mem8(SP+2).bp3-0 abs20(label)+H→PC	---	---	---	---	9	9		0011	1101	1011	000B	bbbH	<abs	20.b	p15~	0..>				*6 *7
	JSRV (tbl4)	SP-3→SP,(PC+3).bp7-0→mem8(SP) (PC+3).bp15-8→mem8(SP+1) (PC+3).H→mem8(SP+2).bp7 0→mem8(SP+2).bp6-4, (PC+3).bp19-16→mem8(SP+2).bp3-0 mem8(x'004080+tbl4<<2)→PC.bp7-0 mem8(x'004080+tbl4<<2+1)→PC.bp15-8 mem8(x'004080+tbl4<<2+2).bp7→PC.H mem8(x'004080+tbl4<<2+2).bp3-0→ PC.bp19-16	---	---	---	---	3	9			1111	1110	<t4>									
NOP	NOP	PC+2→PC	---	---	---	---	2	1	○		0000	0000										

*1 d7 sign-extension
*2 d11 sign-extension
*3 d12 sign-extension
*4 d16 sign-extension
*5 aa=abs18.17 - 16
*6 B=abs20.19
*7 bbb=abs20.18 - 16

MN101E SERIES INSTRUCTION SET

Group	Mnemonic	Operation	Flag				Code Size	Cycle	Re-peat	Exten-sion	1	2	3	4	5	6	7	8	9	10	11	Notes
			VF	NF	CF	ZF																
RTS	RTS	mem8(SP)→(PC).bp7-0 mem8(SP+1)→(PC).bp15-8 mem8(SP+2).bp7→(PC).H mem8(SP+2).bp3-0→(PC).bp19-16 SP+3→SP	---	---	---	---	2	7			0000	0001										
RTI	RTI	mem8(SP)→PSW mem8(SP+1)→(PC).bp7-0 mem8(SP+2)→(PC).bp15-8 mem8(SP+3).bp7→(PC).H mem8(SP+3).bp3-0→(PC).bp19-16 mem8(SP+4)→HA-l mem8(SP+5)→HA-h SP+6→SP	●	●	●	●	2	11			0000	0011										

Contorl instructions

REP	REP imm3	imm3-1→RPC	---	---	---	---	3	2		0010	0001	1rep										*1
BE	BE	PSW & x'3F'→PSW	---	---	---	---	3	3		0010	0010	0000										
BD	BD	PSW | x'c0'→PSW	---	---	---	---	3	3		0010	0011	0000										

*1 no repeat whn imm3=0, (rep: imm3-1)

Other than the instruction of MN101E Series,the assembler of this Series has the following instructions as macro instructions.
The assembler will interpret the macro instructions below as the assembler instructions.

macro instructions	replaced instructions	remarks
INC Dn	ADD 1,Dn	
DEC Dn	ADD -1,Dn	
INC An	ADDW 1,An	
DEC An	ADDW -1,An	
INC2 An	ADDW 2,An	
DEC2 An	ADDW -2,An	
CLR Dn	SUB Dn,Dm	n=m
ASL Dn	ADD Dn,Dm	n=m
LSL Dn	ADD Dn,Dm	n=m
ROL Dn	ADDC Dn,Dm	n=m
NEG Dn	NOT Dn ADD 1,Dn	
NOPL	MOVW DWn,DWm	n=m
MOV (SP),Dn	MOV (0,SP),Dn	
MOV Dn,(SP)	MOV Dn,(0,SP)	
MOVW (SP),DWn	MOVW (0,SP),DWn	
MOVW DWn,(SP)	MOVW DWn,(0,SP)	
MOVW (SP),An	MOVW (0,SP),An	
MOVW An,(SP)	MOVW An,(0,SP)	

Ver3.3(2002.01.31)

17.3 Instruction Map

MN101E SERIES INSTRUCTION MAP

1st nibble\2nd nibble	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NOP	RTS	MOV #8,(io8)	RTI	CMP #8,(abs8)/(abs12)		POP An		ADD #8,Dm				MOVW #8,DWm		MOVW #8,Am	
1	JSR d12(label)		JSR d16(label)		MOV #8,(abs8)/(abs12)		PUSH An		OR #8,Dm				AND #8,Dm			
2	When the exension code is b'oo10'															
3	When the extension code is b'0011'															
4	MOV (abs12),Dm				MOV (abs8),Dm				MOV (An),Dm							
5	MOV Dn,(abs12)				MOV Dn,(abs8)				MOV Dn,(Am)							
6	MOV (io8),Dm				MOV (d4,SP),Dm				MOV (d8,An),Dm							
7	MOV Dn,(io8)				MOV Dn,(d4,SP)				MOV Dn,(d8,Am)							
8	ADD #4,Dm				SUB Dn,Dn				BGE d7	BRA d7	BEQ d7	BNE d7	BCC d7	BCS d7	BLT d7	BLE d7
9	BEQ d4		BNE d4		MOVW DWn,(HA)		MOVW An,(HA)		BGE d11	BRA d11	BEQ d11	BNE d11	BCC d11	BCS d11	BLT d11	BLE d11
A	MOV Dn,Dm / MOV #8,Dm															
B	BSET (abs8)bp								BCLR (abs8)bp							
C	CMP #8,Dm				MOVW (abs8),Am		MOVW (abs8),DWm		CBEQ #8,Dm,d7				CMPW #16,DWm		MOVW #16,DWm	
D	MOV Dn,(HA)				MOVW An,(abs8)		MOVW DWn,(abs8)		CBNE #8,Dm,d7				CMPW #16,Am		MOVW #16,Am	
E	MOVW (An),DWm				MOVW (d4,SP),Am		MOVW (d4,SP),DWm		POP Dn				ADDW #4,Am		BRA d4	
F	MOVW DWn,(Am)				MOVW An,(d4,SP)		MOVW DWn,(d4,SP)		PUSH Dn				ADDW #8,SP	ADDW #4,SP	JSRV (tbl4)	

Extension code: b'0010'

2nd nible\ 3rd nibble	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	MOVW An,Am				CMPW An,Am				MOVW SP,Am		MOVW An,SP		BTST #8,Dm			
1	JMP (A0)	JSR (A0)	JMP (A1)	JSR (A1)	MOV PSW,Dm				REP #3							
2	BE	BGT d7	BHI d7	BLS d7	BNC d7	BNS d7	BVC d7	BVS d7	NOT Dn				ROR Dn			
3	BD	BGT d11	BHI d11	BLS d11	BNC d11	BNS d11	BVC d11	BVS d11	ASR Dn				LSR Dn			
4	SUBW DWn,DWm				SUBW #16,DWm		SUBW #16,Am		SUBW DWn,Am				MOVW DWn,Am			
5	ADDW DWn,DWm				ADDW #16,DWm		ADDW #16,Am		ADDW DWn,Am				CMPW DWn,Am			
6	MOV (d16,SP),Dm				MOV (d8,SP),Dm				MOV (d16,An),Dm							
7	MOV Dn,(d16,SP)				MOV Dn,(d8,SP)				MOV Dn,(d16,Am)							
8	MOVW DWn,DWm (NOPL @n=m)				CMPW DWn,DWm				ADDUW Dn,Am							
9	EXT Dn,DWm		AND #8,PSW	OR #8,PSW	MOV Dn,PSW				ADDSW Dn,Am							
A	SUB Dn,Dm / SUB #8,Dm															
B	SUBC Dn,Dm															
C	MOV (abs16),Dm				MOVW (abs16),Am		MOVW (abs16),DWm		CBEQ #8,Dm,d12				MOVW An,DWm			
D	MOV Dn,(abs16)				MOVW An,(abs16)		MOVW DWn,(abs16)		CBNE #8,Dm,d12				CBEQ #8,(abs8),d7/d11		CBNE #8,(abs8),d7/d11	
E	MOVW (d16,SP),Am		MOVW (d16,SP),DWm		MOVW (d8,SP),Am		MOVW (d8,SP),DWm		MOVW (An),Am				ADDW #8,Am		DIVU	
F	MOVW An,(d16,SP)		MOVW DWn,(d16,SP)		MOVW An,(d8,SP)		MOVW DWn,(d8,SP)		MOVW An,(Am)				ADDW #16,SP		MULU	

Extension code: b'0011'

2nd nibble\ 3rd nibble	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	TBZ (abs8)bp,d7								TBZ (abs8)bp,d11							
1	TBNZ (abs8)bp,d7								TBNZ (abs8)bp,d11							
2	CMP Dn,Dm															
3	ADD Dn,Dm															
4	TBZ (io8)bp,d7								TBZ (io8)bp,d11							
5	TBNZ (io8)bp,d7								TBNZ (io8)bp,d11							
6	OR Dn,Dm															
7	AND Dn,Dm															
8	BSET (io8)bp								BCLR (io8)bp							
9	JMP abs18(label)								JSR abs18(label)							
A	XOR Dn,Dm / XOR #8,Dm															
B	ADDC Dn,Dm															
C	BSET (abs16)bp								BCLR (abs16)bp							
D	BTST (abs16)bp								cmp #8,(abs16)	mov #8,(abs16)	JMP abs20(label)	JSR abs20(label)	CBEQ #8,(abs16),d7/11		CBNE #8,(abs16),d7/11	
E	TBZ (abs16)bp,d7								TBZ (abs16)bp,d11							
F	TBNZ (abs16)bp,d7								TBNZ (abs16)bp,d11							

Ver2.1(2001.03.26)

Record of Changes

Details of revision from Ver.1.1 to Ver.1.2 in MN101EFA8/A7/A3/A2 Series LSI User’s Manual is shown below.

According to the details of revision, “Definition” of the table below is classified into seven groups.

Revision concerning descriptions in LSI User’s Manual:
Writing error correction / Description change / Description addition / Description deletion
Revision concerning LSI specifications:
Specification change / Specification addition / Specification deletion

Modification(Ver.1.2)			Definition	Details of revision	
Page	Title	Line		Ver.1.1	Ver.1.2
I-8 to I-9	Serial Interface:	-	Writing error correction	arbitrary sizes of 1 to 8 bits are selectable.	arbitrary sizes of 2 to 8 bits are selectable.
I-38	Subsection 1.5.7	-	Specification change	-	Alll of "Flash EEPROM Program Conditions" is modified.
I-40	Section 1.6	-	Writing error correction	Package code: LQFP080-P-1414A	Package code: LQFP080-P-1414E
I-41	Section 1.6	-	Writing error correction	Package code: TQFP064-P-1010C	Package code: TQFP064-P-1010D
I-42	Section 1.6	-	Writing error correction	Package code: LQFP064-P-1414A	Package code: LQFP064-P-1414
I-50	Subsection 1.7.6	-	Description deletion	-	Subsection 1.7.6 is deleded.
II-18	Figure:2.2.3	-	Writing error correction	-	BANK1 and BANK2 are added
II-19	Figure:2.2.4	-	Writing error correction	-	BANK1 and BANK2 are added
II-19	Figure:2.2.4	-	Writing error correction	-	Mirroe area size is modified.
II-25	Table:2.2.3	03E66	Writing error correction	TS0ATMAP1L	TS0ATREGAP
II-25	Table:2.2.3	03E6E	Writing error correction	TS1ATMAP1L	TS1ATREGAP
II-25	Table:2.2.3	03EBF	Writing error correction	SELLUD2	SELUD2
II-25	Table:2.2.3	03F57	Writing error correction	SC4STR	SC4STR1
II-27	Table:2.2.4	03E66	Writing error correction	TS0ATMAP1L	TS0ATREGAP
II-27	Table:2.2.4	03EE1	Description deletion	LEDSEL	-
II-27	Table:2.2.4	03EBF	Writing error correction	SELLUD2	SELUD2
II-27	Table:2.2.4	03F57	Writing error correction	SC4STR	SC4STR1

Modification(Ver.1.2)			Definition	Details of revision	
Page	Title	Line		Ver.1.1	Ver.1.2
II-29	Table:2.2.5	03EBF	Writing error correction	SELLUD2	SELUD2
II-29	Table:2.2.5	03F57	Writing error correction	SC4STR	SC4STR1
II-31	Table:2.2.6	03E6X	Description deletion	TS0ATCNT0,TS0ATCNT1,TS0ATTRC1,TS0ATMAP0L,TS0ATMAP0M,,TS0ATMAP1L	-
II-31	Table:2.2.6	03EBF	Writing error correction	SELLUD2	SELUD2
II-31	Table:2.2.6	03F57	Writing error correction	SC4STR	SC4STR1
II-31	Table:2.2.6	03EE1	Description deletion	LEDSEL	-
II-48	First Note	-	Writing error correction	... wait cycle required for high-speed frequency, ...	... wait cycle required for high-speed/low-speed frequency, ...
III-3	Figure:3.1.1	-	Writing error correction	-	Figure:3.1.1 is modified.
III-6	Figure:3.1.2	-	Writing error correction	Low-speed clock for peripheral functions (fslow)	Low-speed clock for peripheral functions (fx)
III-18	Table:3.2.4	-	Writing error correction	10 MHz ≤ fs ≤ 16 MHz	10 MHz ≤ fs ≤ 20 MHz
III-26	Fourth Note	-	Description deletion	In-circuit emulator cannot stop internal flash memory. Debug the function to select internal flash memory operation (operating/stop) under on-board debugging environment.	-
IV-33	TS0CICR	-	Writing error correction	TS0CICR: 0x03FED, TS1CICR: 0x03FED	TS0CICR: 0x03FE9, TS1CICR: 0x03FED
V	Block Diagram	-	Description addition	-	Port 0, 2 to 9, A, B block diagram is added.
V-5	Table:5.2.1	bp2	Writing error correction	P9PLU, 0x03EA9, R/W, Port 9 pull-up resistor control register	P9PLUD, 0x03EA9, R/W, Port 9 pull-up/pull-down resistor control register
V-17	Subsection 5.5.1	-	Writing error correction	the flag of P4IMD is set to "0" to be used as the general port.	the flag of P3IMD is set to "0" to be used as the general port.
V-20	Port 3 Output Mode Register	-	Description deletion	-	"Port 3 Output Mode Register" is deleted.
V-47	Port 5 Output Mode Register	-	Writing error correction	P5OMD7 1: BUZZER P5OMD6 1: NBUZZER	P5OMD7 1: BUZZERA P5OMD6 1: NBUZZERA
V-71	Port 6 Output Mode Register	bp2	Writing error correction	I/O port or special function selection 0: P62 1: TM3IOB	I/O port or special function selection 0: P62 1: TM1IOB
V-80	P7ODC	-	Writing error correction	P7ODC: 0x03EF6	P7ODC: 0x03EF7

Modification(Ver.1.2)			Definition	Details of revision	
Page	Title	Line		Ver.1.1	Ver.1.2
V-87	Port 8 Output Mode Register	-	Description deletion	Port 8 Output Mode Register(P8OMD1:0x03EB8)	Port 8 Output Mode Register(P8OMD:0x03EB8)
V-87	Port 8 Output Mode Register	-	Writing error correction	P8OMD7 1: BUZZER P8OMD6 1: NBUZZER	P8OMD7 1: BUZZERB P8OMD6 1: NBUZZERB
V-90	Subsection 5.11.1	-	Writing error correction	Each bit can be set individually if pull-up resistor is added or not, by P9PLU register. Set the control flag of P9PLU register to "1" to add pull-up resistor.	Each bit can be set individually if pull-up resistor is added or not by P9PLUD register. Set the control flag of P9PLUD register to "1" to add pull-up or pull-down resistor Port 9 can be selected to add pull-up or pull-down register by the SELUD9 flag of SELUD2 register.
V-95	P9PLUD	-	Writing error correction	Port 9 Pull-up/pull-down Resistor Control Register (P9PLU: 0x03EA9)	Port 9 Pull-up/pull-down Resistor Control Register (P9PLUD: 0x03EA9)
V-97	SELUD2	-	Writing error correction	MN101EFA7/MN101EFA2	MN101EFA8/MN101EFA3
VI-6	Figure:6.1.3	-	Writing error correction	Figure:6.1.3 Timer 2 Block Diagram	Figure:6.1.3 Timer 2 and Timer 3 Block Diagram
VI-23	1st Note	-	Writing error correction	If the low-speed clock (fslow) is selected as a count clock source, ...	If the low-speed clock (fx) is selected as a count clock source, ...
VI-23	1st Note	-	Writing error correction	Selecting the synchronous low-speed clock (fslow) as a count clock source solves those problems	Selecting the synchronous low-speed clock (fx) as a count clock source solves those problems
VI-27	Second Note		Writing error correction	-	Table is modified
VI-43	16-bit Timer Cascade Connection Operation	-	Writing error correction	8-bit Timer Cascade Connection Operation (Timer 0 + Timer 1, Timer 2 + Timer 3)	16-bit Timer Cascade Connection Operation (Timer 0 + Timer 1, Timer 2 + Timer 3)
VI-48	Table:6.10.3	-	Writing error correction	fs/8 fslow TM0IO input Synchronous fslow	fs/8 fx TM0IO input Synchronous fx
VI-48	Table:6.10.3	-	Writing error correction	fpll_div: Machine clock (High speed oscillation for peripheral functions) fslow: Machine clock (Low speed oscillation for peripheral functions) fs: System clock	fpll_div: Machine clock (High speed oscillation for peripheral functions) fx: Machine clock (Low speed oscillation for peripheral functions) fs: System clock
VIII-2	1st Note	-	Writing error correction	if there is not much difference in the function between Pin A and C, "A" and "B" of the pin names are omitted.	if there is not much difference in the function between Pin A and B, "A" and "B" of the pin names are omitted.

Modification(Ver.1.2)			Definition	Details of revision	
Page	Title	Line		Ver.1.1	Ver.1.2
VIII-8	Table:8.2.1	Timer 8	Description addition	-	TMCKSEL2, 0x03FB1 TMINSEL2, 0x03FB3
VIII-18	Table:8.3.2	Timer 8	Description addition	Table:8.3.2 Clock Source at Timer Operation (Timer 7)	Table:8.3.2 Clock Source at Timer Operation (Timer 7 and Timer 8)
VIII-39	Description	(2)	Writing error correction	Set the PAOMD6 flag of PAOMD register to "1" to set PA6 pin as a special function pin. Set the PADIR6 flag of PADIR register to "1" to set the output mode. [Chapter 5 I/O Port]	Set the PAOMD5 flag of PAOMD register to "1" to set PA5 pin as a special function pin. Set the PADIR5 flag of PADIR register to "1" to set the output mode. [Chapter 5 I/O Port]
VIII-39	Setup Procedure	(2)	Writing error correction	Set the special function pin to output PAOMD (0x03EBA) bp6: PAOMD6 =1 PADIR (0x03E9A) bp6: PADIR6 =1	Set the special function pin to output PAOMD (0x03EBA) bp5: PAOMD5 =1 PADIR (0x03E9A) bp5: PADIR5 =1
IX-20	PWM Pin Protection Control Register Lower 8 bits	PRTANU1 to 0 PRTAU1 to 0	Writing error correction	TM9OD1/0 output protection function 00: Setting prohibited (When pin protection is used.) 01: Hi-Z output 10: Inactive output 11: Protection is not used	TM9OD1/0 output protection function 00: Unused 01: Hi-Z output 10: Inactive output 11: Setting prohibited
IX-20	First Note	-	Writing error correction	To use the pin protection function, be sure to set up to all the pins. Select "Protection is not used" also to the pins which are not used in this function.	Be sure to set the IRQSEL2 to IRQSEL0 flags before setting the other flags in PWMOFFL register.
IX-21	PWM Pin Protection Control Register Upper 8 bits	PRTANW1 to 0 PRTAW1 to 0 PRTANV1 to 0 PRTAV1 to 0	Writing error correction	TM9OD5 to TM9OD2 output protection function 00: Setting prohibited (When pin protection is used.) 01: Hi-Z output 10: Inactive output 11: Protection is not used	TM9OD5 to TM9OD2 output protection function 00: Unused 01: Hi-Z output 10: Inactive output 11: Setting prohibited
IX-40	Setting PWM Period	Table	Writing error correction	Count clock period × (PWMSETn set value +1) × 2 Count clock period × (PWMSETn set value +1)	Count clock period × (PWMSET set value +1) × 2 Count clock period × (PWMSET set value +1)
IX-48	Table:9.6.7	-	Writing error correction	External interrupt IRQ01 is selected	External interrupt IRQ1 is selected
XI-7	Second Note	-	Description deletion	[Refer to 5.3.3 Points in an interrupt generation]	-
XI-8	Subsection 11.2.3	-	Writing error correction	This LSI allocates 0x04C1 of memory area as flash option areas.	This LSI allocates 0x040C1 of memory area as flash option areas.
XI-11	Table:11.3.1		Writing error correction	-	SLOW is added

Modification(Ver.1.2)			Definition	Details of revision	
Page	Title	Line		Ver.1.1	Ver.1.2
XII-2	Section 12.1	-	Writing error correction	MN101EFA8/A3 has a buzzer. It can output the square wave that multiply by 1/29 to 1/214 of the high frequency oscillation clock. MN101EFA7/A2 does not have a buzzer function.	This LSI has a buzzer. It can output the square wave that multiply by 1/29 to 1/214 of the high frequency oscillation clock.
XII-4	Figure:12.1.1	-	Writing error correction	px	fx
XIII-4	Table:13.1.4	-	Writing error correction	Specification of transfer bit count (1 to 8 bits)	Specification of transfer bit count (2 to 8 bits)
XIII-6	Table:13.1.6	-	Writing error correction	1 to 8 bits (in master communication)	2 to 8 bits (in master communication)
XIII-7	Figure:13.1.1	-	Writing error correction	-	Figure:12.1.1 is modified.
XIII-8	Figure:13.1.2	-	Writing error correction	-	Figure:12.1.2 is modified.
XIII-9	Figure:13.1.3	-	Writing error correction	-	Figure:12.1.3 is modified.
XIII-10	Figure:13.1.4	-	Writing error correction	-	Figure:12.1.4 is modified.
XIII-22	First Note	-	Writing error correction	To change setting values of mode registers, set forced reset first for serial interface by setting both SCnSBIS and SCnSBOS flags of SCnMD1 to "0". (n = 0 to 1)	To change setting values of mode registers, set forced reset first for serial interface by setting both SCnSBIS and SCnSBOS flags of SCnMD1 to "0". (n = 0 to 2)
XIII-23	Second Note	-	Writing error correction	To change setting values of mode registers, set forced reset for serial interface at first by setting both SCnSBIS and SCnSBOS flags of SCnMD1 to "0".	To change setting values of mode registers, set forced reset for serial interface at first by setting both SCnSBIS and SCnSBOS flags of SCnMD1 to "0" (n=0 to 2).
XIII-27	Clock Synchronous Serial Reception Interrupt Control Register	-	Writing error correction	Serial 4(2,1,0) reception interrupt generation timing at clock synchronous serial data	SC4STPCIRQ(SC2RIRQ to SC0RIRQ) generation timing at clock synchronous serial data
XIII-27	Clock Synchronous Serial Reception Interrupt Control Register	-	Writing error correction	SC4IGC 0: Reception is completed	SC4IGC 0: Not used
XIII-38	Transfer Bit Setup	-	Writing error correction	The transfer bit count can be selected from 1 to 8 bits.	The transfer bit count can be selected from 2 to 8 bits.

Modification(Ver.1.2)			Definition	Details of revision	
Page	Title	Line		Ver.1.1	Ver.1.2
XIII-39	Transmit Bit Count and First Transfer Bit	-	Writing error correction	In transmission, when the transfer bit is 1 bit to 7 bits, data storage method for the transmission data buffer differs depending on the first transfer bit specification.	In transmission, when the transfer bit is 2 bit to 7 bits, data storage method for the transmission data buffer differs depending on the first transfer bit specification.
XIII-40	Reception Bit Count and First Transfer Bit	-	Writing error correction	In reception, when the transfer bit is 1 to 7 bits, data storage method for the reception data buffer RXBUFn differs depending on the first transfer bit specification.	In reception, when the transfer bit is 2 to 7 bits, data storage method for the reception data buffer RXBUFn differs depending on the first transfer bit specification.
XIII-40	First Note	-	Writing error correction	When the reception transfer bit count shows 1 to 7 bits, data except the reception data of the specified transfer bit is undefined. The reception data should be masked to use with and instruction or others.	When the reception transfer bit count shows 2 to 7 bits, data except the reception data of the specified transfer bit is undefined. The reception data should be masked to use with and instruction or others.
XIII-41	Continuous Communication	-	Description deletion	When the SCnCTM flag of SCnMD0 register is "1", ... , CPU needs to be activated with a faster system clock.	-
XIII-41	First Note	-	Description change	In continuous communication mode, data should be written into the transmission data buffer to keep continuous communication in a clock slave side. Setting delay causes improper communication since input clock is masked.	When the slave reception is performed with the start condition "enable" in the continuous communication, the system configuration is needed to notify the master of the notification, the data which is read previously may be overwritten.
XIII-41	Second Note	-	Description deletion	-	Second Note is deleted
XIII-41	Third Note	-	Description deletion	-	Third Note is deleted
XIII-52	Recovery from STANDBY Mode by Reception Interrupt	-	Writing error correction	On Serial Interface 0, 1 and 2, a Serial n UART reception interrupt (SCnRIRQ), Serial Interface 4 a Serial 4 Stop condition interrupt (SC4SIRQ) is generated at the falling edge of the serial clock I/O pin (SBTn) by setting the SCnIGC flag of SCINTSEL register to "1".	On Serial Interface 0, 1 and 2, a Serial n UART reception interrupt (SCnRIRQ), Serial Interface 4 a Serial 4 Stop condition interrupt (SC4STPCIRQ) is generated at the falling edge of the serial clock I/O pin (SBTn) by setting the SCnIGC flag of SCINTSEL register to "1".
XIII-58	Synchronous Serial Interface 2 Pin Setup	-	Description addition	-	Synchronous Serial Interface 2 Pin Setup

Modification(Ver.1.2)			Definition	Details of revision	
Page	Title	Line		Ver.1.1	Ver.1.2
XIII-59	Synchronous Serial Interface 4 Pin Setup	-	Description deletion	P0	-
XIII-61	Setup Procedure	(4)	Writing error correction	Control the pin style [set the pin corresponding to each serial] P0ODC(0x03EF0) bp3: P3ODC4 =1 bp4: P3ODC6 =1 P0PLU(0x03EA0) bp3: P3PLU4 =1 bp4: P3PLU6 =1	Control the pin style [set the pin corresponding to each serial] P0ODC(0x03EF0) bp3: P0ODC3 =1 bp4: P0ODC4 =1 P0PLU(0x03EA0) bp3: P0PLU3 =1 bp4: P0PLU4 =1
XIII-64	Description	(1)	Writing error correction	Set the SC1PSCE flag of SC1MD3 register to "1" to select "Enable count" for prescaler count control.	Set the SC0PSCE flag of SC0MD3 register to "1" to select "Enable count" for prescaler count control.
XIII-64	Description	(3)	Writing error correction	Set the SC1FDC1 to 0 flags of SC1MD3 register to "0, 0" to select output fixed at "High" after the SBO1 last data output.	Set the SC0FDC1 to 0 flags of SC0MD3 register to "0, 0" to select output fixed at "High" after the SBO0 last data output.
XIII-64	Description	(2)	Writing error correction	Set the SC1PSC2 to 0 flags of SC1MD3 register to "100" to select fs/2 to the clock source.	Set the SC0PSC2 to 0 flags of SC0MD3 register to "100" to select fs/2 to the clock source.
XIII-64	Description	(4)	Writing error correction	Set the P3ODC4 flags of P0ODC register to "1" to select Nch open-drain to SBT1. Set the P3PLU4 flag of P0PLU register to "1" to enable the pull-up resistor. (Set the pin corresponding to each serial interface)	Set the P0ODC4 flags of P0ODC register to "1" to select Nch open-drain to SBT0. Set the P0PLU4 flag of P0PLU register to "1" to enable the pull-up resistor. (Set the pin corresponding to each serial interface)
XIII-65	Setup Procedure	(6)	Writing error correction	Set SC1MD0 register	Set SC0MD0 register
XIII-65	Description	(6)	Writing error correction	Set the SC0LNG2 to 0 flags of Serial Interface 1 mode register 0 (SC0MD0) to "111" to set the transfer bit count to 8 bits.	Set the SC0LNG2 to 0 flags of Serial Interface 0 mode register 0 (SC0MD0) to "111" to set the transfer bit count to 8 bits.
XIII-66	Description	(13)	Writing error correction	The Serial Interface 1 interrupt SC1TIRQ is generated at the same time of the 8th bits data reception. CPU is then returned from STOP mode to NORMAL mode after the oscillation stabilization wait.	The Serial Interface 1 interrupt SC0TIRQ is generated at the same time of the 8th bits data reception. CPU is then returned from STOP mode to NORMAL mode after the oscillation stabilization wait.
XIII-68	Data Input Pin Setup	-	Writing error correction	Set communication mode by the SCnIOM flag of SC0MD1 register.	Set communication mode by the SCnIOM flag of SCnMD1 register.
XIII-75	Table:13.4.9	-	Writing error correction	300,960,1200,2400,4800	9600,19200,28800,31250,38400

Modification(Ver.1.2)			Definition	Details of revision	
Page	Title	Line		Ver.1.1	Ver.1.2
XIII-81	UART Serial Interface 2 Pin Setup	-	Description addition	-	UART Serial Interface 2 Pin Setup
XIII-83	Setup Procedure	(4)	Description deletion	Select the start condition SC0MD0 (0x03F11) bp3: SC0STE =1	-
XIII-83	Description	(4)	Description deletion	Set the SC0STE flag of SC0MD0 register to "1" to enable start condition.	-
XIII-84	Setup Procedure	(6)	Description addition	bp5: SC0SBIS =1	bp5: SC0SBIS =1 bp6: SC0SBTS =0
XIII-84	Description	(6)	Writing error correction	Set the SC0SBOS and SC0SBIS flags of SC0MD1 register to "1" to set TXD0 pin to serial data output and RXD0 pin to serial data input.	Set the SC0SBOS and SC0SBIS flags of SC0MD1 register to "1" and the SC0SBTS flag of SC0MD1 register to "0" to set TXD0 pin to serial data output and RXD0 pin to serial data input and SBT0 pin to port.
XIII-87	Table:13.5.1	Master/ slave communication	Description addition	500 kHz or more	500 kHz or more, 800 kHz or less
XIII-89	Stop Condition Generation	-	Writing error correction	Stop condition is formed if the data bus (SDA) changes from "Low" to "High" when the clock bus (SCL) is "High".	Stop condition is formed if the data bus (SDA4 pin) changes from "Low" to "High" when the clock bus (SCL4 pin) is "High".
XIII-90	Start/Restart Condition Detection	-	Writing error correction	If the data bus (SDA) changes from "High" to "Low" when the clock bus (SCL) is "High", a start condition is detected and the SC4STRT flag and SC4BUSBSY flag of SC4STR1 register are set to "1".	If the data bus (SDA4 pin) changes from "High" to "Low" when the clock bus (SCL4 pin) is "High", a start condition is detected and the SC4STRT flag and SC4BUSBSY flag of SC4STR1 register are set to "1".
XIII-90	Stop Condition Detection	-	Writing error correction	If the data bus (SDA) changes from "Low" to "High" while the clock bus (SCL) is "High", a start condition is detected. SC4STPCIRQ is generated and the SC4BUSBSY flag of SC4STR1 register is cleared.	If the data bus (SDA4 pin) changes from "Low" to "High" while the clock bus (SCL4 pin) is "High", a start condition is detected. SC4STPCIRQ is generated and the SC4BUSBSY flag of SC4STR1 register is cleared.

Modification(Ver.1.2)			Definition	Details of revision	
Page	Title	Line		Ver.1.1	Ver.1.2
XIII-90	Communication Data Instability Detection	-	Writing error correction	In the case of the master communication, the communication completion interrupt SC4TIRQ is generated after completing 1 byte communication. ... On the other hand, in slave communication, the SC4TIRQ is generated immediately and the communication is completed.	In the case of the master communication, the communication completion interrupt SC4IRQ is generated after completing 1 byte communication. ... On the other hand, in slave communication, the SC4IRQ is generated immediately and the communication is completed.
XIII-92	Reception of Acknowledgement (ACK bit) after Data Transmission	-	Writing error correction	If ACK bit is enabled, data (1 to 8 bits) is transmitted and ACK bit is received from the data receiver.	If ACK bit is enabled, data (2 to 8 bits) is transmitted and ACK bit is received from the data receiver.
XIII-104	Setup Procedure	(4)	Writing error correction	Control the pin direction. P7DIR(0x03E97) bp1: P0DIR6 =1 bp2: P0DIR7 =1	Control the pin direction. P7DIR(0x03E97) bp1: P7DIR1 =1 bp2: P7DIR2 =1
XIII-104	Setup Procedure	(3)	Writing error correction	Control the pin type P7ODC(0x03EF7) bp1: P0ODC6 =1 bp2: P0ODC7 =1 P7PLU(0x03EA7) bp1: P0PLU1 =1 bp2: P0PLU2 =1	Control the pin type P7ODC(0x03EF7) bp1: P7ODC1 =1 bp2: P7ODC2 =1 P7PLU(0x03EA7) bp1: P7PLU1 =1 bp2: P7PLU2 =1
XIII-105	Setup Procedure	(9)	Writing error correction	Set the interrupt level PSW bp6: MIE =0 PERIILR(0x03FFC) bp7 to 6: PERILV1 to 0 =10	Set the interrupt level PSW bp6: MIE =0 PERIILR(0x03FFE) bp7 to 6: PERILV1 to 0 =10
XIII-107	Setup Procedure	(3)	Writing error correction	Control the pin type P7ODC(0x03EF7) bp1: P7ODC1 = 1 bp2: P7ODC2 = 1 P7PLU(0x03EA0) bp1: P7PLU1 = 1 bp2: P7PLU2 = 1	Control the pin type P7ODC(0x03EF7) bp1: P7ODC1 = 1 bp2: P7ODC2 = 1 P7PLU(0x03EA7) bp1: P7PLU1 = 1 bp2: P7PLU2 = 1
XIII-109	Setup Procedure	(13)	Writing error correction	Check the data transmission/reception SC4STR1(0x03F57) bp7: SC4WRS = 0	Check the data transmission/reception SC4STR1(0x03F57) bp7: SC4WRS = 1
XIII-109	Description	(13)	Writing error correction	... by verifying that the SC4WRS flag of SC4STR1 register is set to "0".	... by verifying that the SC4WRS flag of SC4STR1 register is set to "1".
XIV-3	Figure:14.1.1	-	Writing error correction	-	Figure:14.1.1 is modified.
XIV-4	Figure:14.1.2	-	Writing error correction	-	Figure:14.1.2 is modified.
XIV-5	Figure:14.1.3	-	Writing error correction	-	Figure:14.1.3 is modified.

Modification(Ver.1.2)			Definition	Details of revision	
Page	Title	Line		Ver.1.1	Ver.1.2
XIV-6	Figure:14.1.4	-	Writing error correction	-	Figure:14.1.4 is modified.
XIV-13	TS0ADCNT	-	Writing error correction	TS0ADCNT: 0x03E25	TS0ADCNT: 0x03DA5
XIV-13	TS0ADCNT	-	Writing error correction	TS0ADCNT	TS0ADEN
XIV-23	Recommended Circuit with the A/D Conversion	-	Writing error correction	Two Notes	Two notes are deleted. "Recommended Circuit with the A/D Conversion" is added.
XV-11	TS0CKMD	bp7 to 4	Writing error correction	-	"-" is added.
XV-12	TS0TCHSEL	-	Writing error correction	MN101EFA8	MN101EFA8/A7
XV-13	TS1TCHSEL	-	Writing error correction	MN101EFA7	MN101EFA8
XV-14	TS0RESULT	-	Writing error correction	MN101EFA8	MN101EFA8/A7
XV-15	TS1RESULT	-	Writing error correction	MN101EFA7	MN101EFA8
XV-16	TS0ERROR	-	Writing error correction	TS0ERROR: 0x03DA3, TS1ERROR: 0x03DD3	TS0ERROR: 0x03DA4, TS1ERROR: 0x03DD4
XV-16	TS0ERROR	-	Writing error correction	MN101EFA8	MN101EFA8/A7
XV-17	TS1ERROR	-	Writing error correction	MN101EFA7	MN101EFA8
XV-18	TSnINm Expected Data Register	bp7 to 4	Writing error correction	-	"-" is added.
XV-19	TSnINm Measurement Data Register (Lower 8 bits)	-	Writing error correction	R/W,R/W,R/W,R/W,R/W,R/W,R/W,R/W	R,R,R,R,R,R,R,R
XV-19	TSnINm Measurement Data Register (Lower 2 bits)	-	Writing error correction	0,0,0,0,0,0,0,0 R,R,R,R,R,R,R/W,R/W	-,-,-,-,-,0,0 -,-,-,-,-,R,R
XV-20	TS0ATCNT0	-	Writing error correction	-	Not implemented flags are modified.
XV-21	TS0ATCNT1	-	Writing error correction	-	Not implemented flags are modified.
XV-22	TSxATRAMAPH	-	Writing error correction	-	Not implemented flags are modified.
XV-28	Second Note	-	Description change	-	It merged Second note and Third note.
XV-30	Second Note	-	Description change	-	It merged Second note and Third note.

Modification(Ver.1.2)			Definition	Details of revision	
Page	Title	Line		Ver.1.1	Ver.1.2
XV-34	Setting of Transfer Address to Memory Pointer	-	Writing error correction	Set each start address of register space and RAM space for data automatic transfer in TSnATREGAP, TSnATRAMAPH and TSATRAMAPL registers.	Set each start address of register space and RAM space for data automatic transfer in TSnATREGAP, TSnATRAMAPH and TSnATRAMAPL registers.
XVI-2	Section 16.1	Line 1	Writing error correction	This LSI includes 64 KB of a flash memory as an internal instruction memory.	MN101EFAxG includes 128KB of a flash memory as an internal instruction memory. MN101EFAxD includes 64KB of a flash memory as an internal instruction memory.
XVI-2	Section 16.1	Line 3	Writing error correction	In this LSI, there are four sectors...	In MN101EFAxG, there are seven sectors: 32 KB of Sector 0 to 2, 20 KB of Sector 3 and 4 KB of Sector 4 to 6. IN MN101EFAxD, there are five sectors ...
XVI-2	Table:16.1.1	Capacity	Writing error correction	64 KB	128 KB (MN101EFAxG) 64 KB (MN101EFAxD)
XVI-2	Table:16.1.1	-	Writing error correction	Number of times of rewriting in each sector, Maximum 1000 times Data retention period, 10 years	Number of times of rewriting in each sector, 32Kbyte, 20Kbyte Sector : Over 1000 times. 4Kbyte Sector : Over 10000 times. Data retention period, 20 years
XVI-4	Table:16.1.2	Rewritable area	Writing error correction	Sector 0 to 4,Sector 0 to 4,Sector 0 to 4,Sector 0 to 4	Sector 0 to 6 (MN101EFAxG),Sector 0 to 6 (MN101EFAxG),Sector 0 to 6 (MN101EFAxG),Sector 0 to 6 (MN101EFAxG) Sector 0 to 4 (MN101EFAxD),Sector 0 to 4 (MN101EFAxD),Sector 0 to 4 (MN101EFAxD),Sector 0 to 4 (MN101EFAxD)
XVI-4	Table:16.1.2	Features	Writing error correction	Sector 4 is used as activating area	Sector 6 (MN101EFAxG) or Sector 4 (MN101EFAxD) are used as activating area
XVI-4	Subsection 16.1.2	Line 2	Writing error correction	Table:16.1.3 shows the memory map of the internal flash memory.	Table:16.1.3 and Table:16.1.4 show the memory map of the internal flash memory.
XVI-4	Table:16.1.3	-	Description addition	-	Table:16.1.3 is added
XVI-5	Table:16.1.4	Sector	Writing error correction	Sector 0, Sector 1, Sector 1, Sector 2, Sector 3(also used as BOOT area)	Sector 0, Sector 1, Sector 2, Sector 3, Sector 4(also used as BOOT area)

Modification(Ver.1.2)			Definition	Details of revision	
Page	Title	Line		Ver.1.1	Ver.1.2
XVI-5	Table:16.1.4	-	Writing error correction	"MAIN area (FBEWER = 0x4B)" included "BOOT area is not used"	"MAIN area (FBEWER = 0x4B)" is modified to include "BOOT area is used"
XVI-5	Table:16.1.4	-	Writing error correction	Sector 0, 20 KB	Sector 1, 20 KB
XVI-6	Section 16.2	-	Writing error correction	For parallel programmer writing, Sector 0 to 4 can be rewritten.	For parallel programmer writing, All sectors can be rewritten.
XVI-7	Section 16.3	-	Writing error correction	For serial programmer writing, Sector 0 to 4 can be rewritten.	For serial programmer writing, All sectors can be rewritten.
XVI-9	Subsection 16.4.1	-	Writing error correction	Except for the V_{DD5} and V_{SS} lines, the line length of the signal from the connector to the microcomputer must be less than 50 cm.	Except for the VDD5 and VSS lines, the line length of the signal from the connector to the microcomputer must be less than 50 cm.
XVI-12	Figure:16.5.1	-	Description addition	-	Figure:16.5.1 is added
XVI-12	Section 16.5	Line 9	Writing error correction	Figure:16.5.1 shows the memory map in the user mode microcontroller rewriting.	Figure:16.5.1 and Figure:16.5.2 show the memory map in the user mode microcontroller rewriting.
XVI-13	Section 16.5	Line 2	Writing error correction	Sector 0 to 4 are addressed in 0x04000 to 0x13FFF, and ...	Sector 0 to 4 are addressed in 0x04000 to 0x23FFF[0x13FFF], and ...
XVI-13	Section 16.5	Line 3	Writing error correction	When 0x4B is set in the rewriting enable register (FBEWER), reserved area is addressed in 0x6F000 to enable command library, and Sector 0 to 4 can be rewritten.	When 0x4B is set in the rewriting enable register (FBEWER), reserved area is addressed in 0x6F000 to enable command library, and Sector 0 to 6 [Sector 0 to 4] can be rewritten.
XVI-17	Figure:16.6.1	-	Description addition	-	Figure:16.6.1 is added
XVI-17	Subsection 16.6.1	Line 3	Writing error correction	Figure:16.6.1 shows the memory map in BOOT mode microcontroller rewriting.	Figure:16.6.1 and Figure:16.6.2 show the memory map in BOOT mode microcontroller rewriting.
XVI-18	Subsection 16.6.1	Line 1	Writing error correction	Sector 4 Sector 0 to 3 0x13000	Sector 6 [Sector 4] Sector 0 to 5 [Sector 0 to 3] 0x23000 [0x13000]
XVI-19	Subsection 16.6.2	Line 1	Description deletion	If BOOT area (Sector 4) is rewritten by mistake, ...	If BOOT area is rewritten by mistake, ...
XVI-19	Subsection 16.6.2	Line 2	Writing error correction	To use Sector 4 as BOOT area, it is recommended to protect Sector 4 in advance.	To use Sector 6 [Sector 4] as BOOT area, it is recommended to protect Sector 6 [Sector 4] in advance.

Modification(Ver.1.2)			Definition	Details of revision	
Page	Title	Line		Ver.1.1	Ver.1.2
XVI-27	Configuration 1	-	Writing error correction	The size of the data file for MAIN area should be adjusted to less than 64 KB.	The size of the data file for MAIN area should be adjusted to less than 128 KB [64 KB].
XVI-27	Configuration 2,3	-	Writing error correction	Sector 4 is used as BOOT area.	Sector 6 [Sector 4] is used as BOOT area.
XVI-27	Configuration 2,3	-	Writing error correction	The size of the data file for MAIN area should be adjusted to less than 60 KB.	The size of the data file for MAIN area should be adjusted to less than 124 KB [60 KB].
XVI-27	Configuration 4	-	Writing error correction	The size of the data file for MAIN area should be adjusted to less than 64 KB.	The size of the data file for MAIN area should be adjusted to less than 128 KB [64 KB].
XVI-27	Configuration 5,6	-	Writing error correction	Sector 4 is used as BOOT area.	Sector 6 [Sector 4] is used as BOOT area.
XVI-27	Configuration 5,6	-	Writing error correction	The size of the data file for MAIN area should be adjusted to less than 60 KB.	The size of the data file for MAIN area should be adjusted to less than 124 KB [60 KB].
XVI-28	Table:16.8.2	-	Description addition	-	Table:16.8.2 is added
XVI-28	Subsection 16.8.2	Line 3	Writing error correction	Table:16.8.2 shows the protect information.	Table:16.8.2 and Table:16.8.3 show the protect information.
XVI-29	Subsection 16.8.2	Line 2	Writing error correction	Compose the file for protect / security of 4 KB + 64 B.	Compose the file for protect / security of 4 KB + 128 KB [64 KB].
XVII-11	0x03EA9	-	Writing error correction	P9PLU4, P9PLU3, P9PLU2, P9PLU1, P9PLU0	P9PLUD4, P9PLUD3, P9PLUD2, P9PLUD1, P9PLUD0

Inquiries

If you have questions regarding technical information on this manual, please visit the following URL.

Panasonic Corporation

URL: http://www.semicon.panasonic.co.jp/en

- Microcomputer Home Page
  http://www.semicon.panasonic.co.jp/e-micom

MN101EFA8/A7/A3/A2 Series
LSI User's Manual

August 31, 2012 1st Edition 2st Printing

Issued by
Panasonic Corporation

Semiconductor Business Group
Industrial Devices Company
Panasonic Corporation

1 Kotari-yakemachi, Nagaokakyo City, Kyoto
617-8520, Japan
Tel : 81-75-951-8151
http://www.semicon.panasonic.co.jp/en

130112
Printed in Japan